



UNIVERSIDAD  
DE MÁLAGA



E.T.S.  
INGENIERÍA  
INFORMÁTICA

ESCUELA SUPERIOR DE INGENIERÍA INFORMÁTICA  
INGENIERÍA DE LA SALUD

---

**SIMULACIÓN DE TEJIDOS CANCERÍGENOS UTILIZANDO  
AUTÓMATAS CELULARES PROBABILÍSTICOS.**

**CANCEROUS TISSUES SIMULATION USING  
PROBABILISTIC CELLULAR AUTOMATA.**

---

**Realizado por:**

Rocío Egea Guerrero

**Tutorizado por:**

Dr. Gabriel Aguilera Venegas y Dr. José Luis Galán García

**Departamento:**

Matemática aplicada (MA)

UNIVERSIDAD DE MÁLAGA

MÁLAGA, JULIO 2017

Fecha de defensa:

El secretario del Tribunal



## Resumen:

El cáncer es una de las principales causas de muerte a nivel mundial. Este término abarca más de doscientos tipos de enfermedades, las cuales se caracterizan por un crecimiento descontrolado de las células y la posible diseminación por el organismo.

El presente trabajo describe el diseño e implementación de un programa basado en el “Juego de la vida” creado por el matemático John Horton Conway, mediante el cual se permite modelar el comportamiento de un tejido que sufre cáncer, en función de los valores dados a una serie de parámetros. Dicho programa implementa una versión más versátil que la propuesta en el “Juego de la vida”, ya que este se basa en un autómata celular probabilístico. Gracias a ello, se amplía la complejidad de los modelos, de forma que pueden llegar a simularse situaciones reales de ciertos tejidos.

Para ello, el trabajo se divide en primer lugar en una introducción donde se recogen los principales aspectos sobre el cáncer y la descripción de autómatas celulares. En segundo lugar, la implementación de un programa diseñado en Java, el cual permite a través de una entrada de datos, modelar ciertos comportamientos que pueda presentar un tejido en diferentes situaciones. Por último, una fase de pruebas y resultados donde estudiando diferentes tejidos arbitrarios en condiciones ideales, poder establecer una relación entre la simulación obtenida y procesos fisiopatológicos que acontecen en el cuerpo humano.

**Palabras clave:** Autómata celular, cáncer, tumor, muerte celular, supervivencia celular, nacimiento celular, simulación, tejido, células vecinas.

## **Abstract:**

Cancer is one of the leading causes of death worldwide. This pathology includes more than two hundred different types of diseases. All of them are characterized by an uncontrolled growth of tumoral cells locally. In some circumstances, they could spread over tissues and organs in the organism.

This research project describes a program based on “Game of life” model which was created by the mathematician John Horton Conway, where by using different specific variables, the software could predict cells behavior. This program shows an implemented and more versatile version of the classic “Game of life” and is based on a probabilistic cellular automata. Thanks to that, the complexity of the models are increased and consequently, the program might simulate specific tissues and organs cells.

To achieve that, this degree final project is divided in different parts. Firstly, an introduction which explains the main aspect of cancer and cellular automata. Second part describes the implementation of a program designed in Java, which allows build different models of cells behavior establishing several parameters. Final chapter depicts a phase of tests and results, which aim is to study different arbitrary tissues under ideal conditions. It allows the establishment of the relation between the obtained simulation and pathophysiological processes that occur in the human body.

**Key words:** Cellular automata, cancer, tumor, cell death, cell survival, cell neogenesis, simulation, tissue, cell surrounding.

# Agradecimientos

*En primer lugar, quisiera dar las gracias a mis tutores del trabajo fin de grado, Dr. Gabriel Aguilera Venegas y Dr. José Luis Galán García, por ayudarme a desarrollar este trabajo, siempre tan positivos, con una sonrisa y mostrando la ilusión y esperanza depositada en este proyecto, dándome fuerzas para seguir avanzando.*

*A mis padres, Juan José y Concha, por ser los padres que todo hijo desearía tener. Gracias por concederme siempre vuestra confianza y apoyo. A mi hermano Julio, por ser tan cariñoso y darme en todo momento tus consejos de hermano mayor. A Juanjo, por ser tan brillante y tener un corazón que no te cabe en el pecho, gracias por tu asesoramiento y conocimientos que tanto me han ayudado a la hora de realizar este trabajo. A Andrés, por todos tus consejos y momentos juntos que no cambiaría por nada en el mundo. Mi familia, mi mejor ejemplo a seguir.*

*También quisiera dar las gracias a Ángeles, por cuidarme con tanto cariño y preocuparse siempre por mí durante todos estos años.*

*Por último, agradecer a todas aquellas personas que me han ayudado y apoyado durante la carrera, ya que sin ellos nada de esto habría sido posible.*

# ÍNDICE GENERAL

<b>Capítulo 1: Introducción.....</b>	<b>13</b>
1.1. Antecedentes.....	13
1.2. Motivación .....	13
1.3. Objetivos .....	14
1.4. Método .....	15
1.5. Herramientas .....	15
1.6. Estructura de la memoria.....	16
<b>Capítulo 2: Marco teórico de la investigación .....</b>	<b>17</b>
2.1. Cáncer.....	17
2.1.1. Introducción.....	17
2.1.2. Células cancerígenas .....	19
2.1.3. Grado de un tumor.....	19
2.1.4. Estadio de un tumor.....	20
2.1.5. Terapias .....	20
2.2. El juego de la vida .....	21
2.2.1. ¿Qué es un autómatas celular?.....	21
2.2.2. Reglas del Juego de la vida.....	22
2.2.3. Patrones .....	24
2.2.3.1. Estáticos.....	25
2.2.3.2. Patrones de crecimiento indefinido .....	25
2.2.3.3. Osciladores .....	25
2.2.3.4. Deslizadores.....	25
2.3. Variantes del Juego de la vida .....	26
<b>Capítulo 3: Diseño del programa .....</b>	<b>29</b>
3.1. Descripción.....	29
3.2. Implementación del programa .....	30
3.2.1. Clases .....	30
3.2.2. Atributos .....	32
3.2.2.1. Descripción de las probabilidades pv y pm. ....	34
3.2.2.2. ¿Cómo se asignan las probabilidades pv y pm? .....	36
3.2.2.3. Descripción probabilidades pvv y pmv .....	36
3.2.3. Métodos.....	37
<b>Capítulo 4: Pruebas y resultados.....</b>	<b>45</b>
4.1. Simulación 1: Juego de la vida original .....	46

4.2.	Simulación 2: Variación de densidad de población .....	47
4.3.	Simulación 3: Variación de reglas.....	52
4.4.	Simulación 4: Variación de reglas y probabilidades. ....	55
4.5.	Simulación 5: Simulación ideal de tejidos con disposición fibrilar.....	60
<b>Capítulo 5: Conclusiones y trabajos futuros. ....</b>		<b>65</b>
5.1.	Conclusiones .....	65
5.2.	Futuras líneas de desarrollo .....	66
<b>Anexo A: Manual de usuario .....</b>		<b>69</b>
A.1.	Instalación y puesta en marcha.....	69
A.2.	Entrada de datos.....	70
A.3.	Interfaz gráfica de resultados .....	75
<b>Anexo B. Interpretación gráfica de la variación de las reglas del autómata celular.....</b>		<b>77</b>
<b>Glosario de términos .....</b>		<b>81</b>
<b>Bibliografía y referencias.....</b>		<b>85</b>

## ÍNDICE DE FIGURAS

<b>Figura 2.1.</b> Niveles de organización celular en seres vivos. ....	17
<b>Figura 2.2.</b> Vecindad de Moore.. ....	22
<b>Figura 2.3.</b> Ejemplo de array bidimensional en el JLV (Primera generación).....	23
<b>Figura 2.4.</b> Representación de la segunda generación de células.....	24
<b>Figura 2.5.</b> Ejemplos de patrones estáticos.....	25
<b>Figura 2.6.</b> Ejemplos de patrones osciladores.....	25
<b>Figura 2.7</b> Ejemplo de patrón deslizador.....	26
<b>Figura 3.1.</b> Diagrama de clases UML. ....	31
<b>Figura 3.2.</b> Ejemplo de matriz de células en estudio. ....	35
<b>Figura 3.4.</b> Diagrama de flujo genérico del método DatosProbabilidad. ....	40
<b>Figura 3.5.</b> Diagrama de flujo del método DatosProbabilidad para el JLV.....	41
<b>Figura 3.6.</b> Diagrama de flujo del método reglasJuego. ....	43
<b>Figura 4.1.</b> Matriz correspondiente a la simulación 1.....	46
<b>Figura 4.2.</b> Simulación de la matriz de la figura 4.1 con el JLV. ....	46
<b>Figura 4.3.</b> Diferencia de densidad de células entre tejido hepático y tejido cartilaginoso .....	47
<b>Figura 4.4.</b> Simulación con un porcentaje de densidad de población del 5% .....	48
<b>Figura 4.5.</b> Porcentaje de densidad de población del 45%.....	49
<b>Figura 4.6.</b> Diferencia entre tomografía axial computarizada celebrar normal y una con reducción de masa cerebral correspondiente a demencia senil .....	50
<b>Figura 4.7.</b> Generación de tejido con una densidad de población del 90%.....	51
<b>Figura 4.8.</b> Pieza de tumor pulmonar con áreas necróticas.....	52
<b>Figura 4.9.</b> Representación de la generación 8 de un mismo tejido con diferentes reglas que rigen el comportamiento celular.....	53
<b>Figura 4.10:</b> Simulación de la generación 5 del tejido empleado en la figura 4.9 con diferentes reglas que rigen el comportamiento celular.....	55
<b>Figura 4.11.</b> Simulación de tejido con crecimiento descontrolado. ....	56
<b>Figura 4.12.</b> Infiltración de masa tumoral .....	57
<b>Figura 4.13.</b> Simulación JLV variando los parámetros pvv y pmv.....	58
<b>Figura 4.14.</b> Representación de la matriz esperada según el JLV y la matriz obtenida al modificar el pvv y pmv. ....	58
<b>Figura 4.15.</b> Representación de un mismo tejido con diferentes parámetros de entrada .....	59
<b>Figura 4.16 :</b> Visión de tendón a través de microscopía óptica.....	60



<b>Figura 4.17:</b> Visión de tejido muscular liso de una sección del intestino delgado a través de microscopía óptica. ....	60
<b>Figura 4.18.</b> Simulación ideal del comportamiento de un tejido fibroso. ....	62
<b>Figura 4.19.</b> Simulación de tejido arbitrario con tendencia diagonal.....	66
<b>Figura A.1.</b> Ejecución del programa mediante Eclipse. ....	69
<b>Figura A.2.</b> Ejecución del programa mediante cmd. ....	70
<b>Figura A.3.</b> Menú inicial del programa.....	70
<b>Figura A.4.</b> Apariencia de la primera opción de entrada de datos .....	71
<b>Figura A.5.</b> Ejemplo demostrativo de opción dos de entrada de datos.....	72
<b>Figura A.6.</b> Fichero vacío generado .....	72
<b>Figura A.7.</b> Fichero de ejemplo .....	73
<b>Figura A.8.</b> Ejemplo de tercera opción del menú de inicio.....	74
<b>Figura A.9.</b> Ejemplo de cuarta opción del menú de inicio .....	74
<b>Figura A.10.</b> Ejemplo de interfaz gráfica .....	75
<b>Figura B.1.</b> Condiciones genéricas que determinan el estado de la célula en estudio para la siguiente generación.....	77
<b>Figura B.2.</b> Representación gráfica de las condiciones que determinan el desarrollo del autómatas celular para el JLV original (Simulación A, figura 4.9). ....	78
<b>Figura B.3.</b> Representación gráfica de las condiciones impuestas en la simulación C de la figura 4.9. ....	78

## ÍNDICE DE TABLAS

<b>Tabla 2.1.</b> Ejemplo de variantes del JLV.....	27
<b>Tabla 3.1.</b> Definición de los atributos más relevantes del programa .....	34
<b>Tabla 4.1.</b> Parámetros empleados en la simulación D de la figura 4.9.....	54
<b>Tabla 4.2.</b> Parámetros empleados para la simulación de la figura 4.10. ....	54
<b>Tabla 4.3.</b> Parámetros empleados para la simulación de la figura 4.11. ....	57
<b>Tabla 4.4.</b> Parámetros empleados para la simulación de la figura 4.18 .....	61
<b>Tabla 4.5.</b> Parámetros empleados para la simulación de la figura 4.19. ....	63

# Capítulo 1: Introducción

## 1.1. Antecedentes

En 1970, el matemático John Horton Conway creó el “Juego de la vida” (JLV). Se trata de un autómata celular, es decir, un modelo matemático para un sistema dinámico que evoluciona en pasos discretos. El juego se desarrolla en una matriz o cuadrícula, teóricamente infinita, que se extiende en todas las direcciones del plano. Cada una de las casillas de este tablero está compuesta por una célula viva o muerta. A su vez, cada célula está rodeada por 8 células vecinas. Como se ha mencionado antes, el juego evoluciona en pasos discretos, dicho de otra forma, por turnos. En cada turno, se examinan todas las casillas del tablero y se calcula cuál será el próximo estado para cada una de las células en función del estadio que presenten sus células vecinas en el momento de la iteración. Esto se realiza mediante tres reglas simples, las cuales provocan el nacimiento, supervivencia o muerte de cada una de las células.

Una vez realizados los cálculos, se actualiza el tablero y se repite el proceso. En el capítulo 2, se realizará una descripción más detallada del JLV.

Este juego ha causado gran interés a lo largo de la historia, debido a que a partir de unas reglas sencillas se puede llegar a patrones muy complejos. Otro de los factores que impulsó su éxito fue la similitud con algunos de los procesos de la vida real.

Se dice que es un juego de 0 jugadores, dado que su evolución está determinada por un estado inicial de las células y no necesita ninguna entrada de datos posterior [1] [2].

## 1.2. Motivación

El cáncer es una de las principales causas de morbilidad y mortalidad en

todo el mundo. Según la Organización Mundial de la Salud (OMS), en 2012 se produjeron unos 14 millones de nuevos casos y 8,2 millones de muertes fueron relacionadas con dicha enfermedad. Se prevé que el número de nuevos casos aumente aproximadamente un 70% en los próximos 20 años [3].

Por este motivo, el trabajo fin de grado (TFG) tiene como objetivo proporcionar información en términos probabilísticos de cómo podría evolucionar un tumor, ya que se trata de un tema de vital importancia hoy en día.

Actualmente, hasta donde nosotros conocemos, no existen publicaciones que relacionen el JLV de John Horton Conway con células cancerígenas. Con este TFG se espera simular el comportamiento de células cancerígenas a través de autómatas celulares probabilísticos dando lugar a un programa mucho más versátil que el original. A partir de los datos iniciales introducidos por el usuario, se podrá conocer la evolución de dichas células. Para ello, se deben de implementar unas nuevas reglas de simulación, así como el uso de distribuciones de probabilidad para el estudio de los diferentes casos. Por lo tanto, se puede enmarcar este TFG como un trabajo de investigación.

Los tutores de este TFG han trabajado anteriormente con autómatas celulares en simulación. Algunas de sus publicaciones están recogidas en el apartado de Bibliografía y referencias [4] [5] [6] [7].

### **1.3. Objetivos**

El objetivo general de este trabajo es la simulación del tejido cancerígeno mediante un modelo matemático conocido como autómata celular. Para ello, será necesario cumplir con una serie de objetivos específicos:

1. Búsqueda bibliográfica de artículos y referencias relacionadas con el JLV, el cáncer, características que tienen estas células y todo tipo de información que pueda ser útil para la realización del trabajo.
2. Creación de unas reglas que cumplan las condiciones que simulen la

reproducción de las células cancerígenas mediante la utilización de distribuciones de probabilidad.

3. Implementación de un programa que nos muestre dicho resultado a partir de una serie de datos que introduzca el usuario.

## **1.4. Método**

El método a emplear para el desarrollo del trabajo consiste en la técnica de prototipado simple. Esta técnica consiste en el reparto del proyecto en distintas fases, de forma que se van implementando de forma evolutiva, teniendo como resultado un prototipo desde la primera etapa del mismo. A medida que se van completando las fases parciales, estas se irán cargando de funcionalidad e irán reemplazando a las que les precedían hasta llegar al resultado final. La principal ventaja que presenta esta metodología es que resulta más sencilla la búsqueda de errores dentro de las etapas realizadas y a su vez proporciona flexibilidad a la hora de la implementación del proyecto.

## **1.5. Herramientas**

Para el desarrollo del trabajo se necesita la ayuda del siguiente material informático:

### **1.5.1. Java**

Se trata de un lenguaje de programación de alto nivel orientado a objetos y basado en clases. El entorno de desarrollo que se utilizará para programar en java será eclipse. Actualmente es uno de los lenguajes más utilizados en programación dado que se puede ejecutar en múltiples plataformas, es portable, robusto y dinámico, con la habilidad de cubrir las diferentes necesidades para cada tipo de aplicación.

La idea original de este trabajo era la utilización de un sistema de álgebra computacional conocido como Maxima. Finalmente se ha decidido no utilizarlo dada la complejidad y a que no aporta grandes ventajas en este caso concreto.

El beneficio que se obtiene al abandonar la conexión de Maxima con Java es más tiempo para la realización del resto de fases del trabajo. Por tanto, finalmente se ha optado por usar solo Java.

## 1.6. Estructura de la memoria

La memoria de este trabajo fin de grado se divide en 5 capítulos, 2 anexos, un glosario de términos y un apartado de bibliografía y referencias. La distribución del contenido se muestra a continuación:

El **capítulo 2: Marco teórico de la investigación**, realiza una descripción breve de los principales aspectos del cáncer y del JLV.

El **capítulo 3: Diseño del programa**, hace una descripción del programa y expone sus principales componentes detallando el funcionamiento de cada uno de ellos.

El **capítulo 4: Pruebas y resultados**, se van a realizar simulaciones con el programa realizado introduciendo distintos tipos de datos para observar y poder determinar cómo se comporta el autómata celular probabilístico. Con las simulaciones obtenidas se buscarán ejemplos que simulen un comportamiento parecido en procesos reales del organismo.

El **capítulo 5: Conclusiones y trabajos futuros**, se tratarán los aspectos más relevantes del trabajo, así como las futuras líneas de desarrollo que puedan derivar de este.

El **Anexo A: Manual de usuario**, trata el modo que un usuario debe de iniciar el programa y su puesta en funcionamiento.

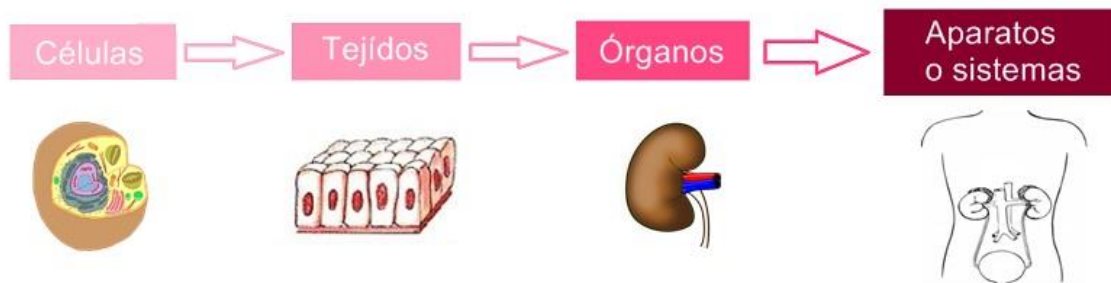
El **Anexo B: Interpretación gráfica de las reglas del autómata celular**, es información complementaria para facilitar al lector la comprensión de algunos conceptos.

# Capítulo 2: Marco teórico de la investigación

## 2.1. Cáncer

### 2.1.1. Introducción

La célula es la unidad fundamental y estructural de todos los organismos vivos. Es la mínima unidad de materia que puede llevar a cabo las funciones vitales de un ser vivo, las cuales son: nutrición, relación y reproducción. Además, esta contiene material genético, el cual dicta las características del ser vivo y es transmitido a la descendencia. El cuerpo humano está compuesto por millones de células. Estas se agrupan en tejidos. Los diferentes tejidos se unen a su vez para construir órganos y el conjunto de órganos se integran en aparatos o sistemas dentro de los seres vivos. Esto puede dar una idea de la importancia que tienen dentro del cuerpo humano y lo primordial que es su correcto funcionamiento.



**Figura 2.1.** Niveles de organización celular en seres vivos.

En condiciones normales, las células trabajan al unísono y están coordinadas entre sí. El organismo está en continuo cambio y periódicamente los tejidos se renuevan y las células envejecidas dejan paso a nuevas células, ya que estas están programadas para saber cuándo deben de reproducirse o morir. En ocasiones, este mecanismo de reproducción puede fallar, dando

lugar a una reproducción descontrolada de células. Esto puede deberse a diversos factores, como, por ejemplo, la intervención de agentes exógenos de tipo físico, químico o biológico, que pueden afectar a los mecanismos reguladores del crecimiento y a la genética de la célula o bien a causa de agentes endógenos de origen genético (mutaciones) o metabólico (hormonas y enzimas) que actúan como agentes favorecedores del crecimiento celular. Las células ante este tipo de acontecimientos tienen un mecanismo de autodestrucción conocido como apoptosis celular, con el fin de erradicar la mutación y no transmitir este material genético defectuoso a la descendencia. Sin embargo, existen ocasiones en las que este mecanismo de defensa falla dando lugar a la conocida enfermedad denominada cáncer.

A partir de una célula cancerígena inicial, la cual se produce por la alteración de algunos de sus genes, se forma una masa de células, denominada tumor, de crecimiento variable que invade los tejidos que rodea o incluso tejidos no colindantes en estados más avanzados. Las células tumorales se suelen caracterizar por la velocidad de proliferación que presentan.

Los tumores se pueden clasificar en dos conjuntos:

1. Tumores homotípicos o benignos (estructura histológica normal) que crecen, pero no matan, salvo raras excepciones.
2. Tumores heterotípicos o malignos, es decir, de evolución grave que pueden provocar en última estancia la muerte. Este puede presentar diversos grados de diferenciación, desde una estructura similar a la normal, pasando por diferentes grados de diferenciación, hasta la pérdida total de la estructura. Estos tumores se caracterizan por poder liberar células malignas a la circulación, las cuales se pueden asentar en otros lugares del cuerpo diferentes al del origen provocando tumores secundarios o metástasis

En forma de resumen, Santiago Ramón y Cajal definió el cáncer como una alteración de la división celular y la proliferación que origina una masa celular



excesiva, sin relación con las necesidades del organismo, que tiende a persistir sin coordinación con el crecimiento ordenado de los tejidos. No es beneficioso para el organismo, vive a sus expensas y conduce a la caquexia y muerte. Esta definición quedaría completa añadiendo las propiedades invasivas y metastásicas de origen plurietiológico, la transformación neoplásica de las células y la activación de oncogenes [8].

### **2.1.2. Células cancerígenas**

La célula, a consecuencia de la transformación neoplásica, sufre una serie de transformaciones. Uno de los factores más relevantes para la distinción de células cancerígenas es su alta velocidad de reproducción. Otro tipo de alteraciones que puede sufrir la célula entre otras son: modificación de la densidad de carga superficial, cambios en la membrana nuclear, cambios en ribosomas, producción de receptores de factores de crecimiento, alteraciones de cromosomas y genes.

### **2.1.3. Grado de un tumor**

Como se ha expuesto anteriormente, un tumor puede tener diferentes grados de diferenciación en sus células. Existen cuatro grados para determinar tumores, dependiendo de lo diferenciadas que estén sus células de las células del tejido normal. En los tumores de grado uno, las células tumorales están poco diferenciadas, es decir, que estas células van a tender a crecer y diseminarse lentamente. Sin embargo, cuanto más alto sea el nivel del grado, más diferenciadas serán las células del tumor y, por lo tanto, tenderán a crecer y difuminarse más rápidamente [9].

Generalmente se usa el siguiente sistema de graduación:

- GX: No es posible determinar un grado
- G1: Células tumorales bien diferenciadas (Grado bajo)
- G2: Células tumorales moderadamente diferenciadas (Grado intermedio)

- G3: Células tumorales escasamente diferenciadas (Grado alto)
- G4: Células tumorales indiferenciadas (Grado máximo)

En resumen, lo definido anteriormente se le conoce como grado histológico en función de la agresividad o malignidad del tumor.

#### **2.1.4. Estadio de un tumor**

Sirve para proporcionar información sobre el tumor. Uno de los sistemas más utilizados para describir el tipo de tumor y su extensión es el sistema TNM, el cual se basa:

- T: Indica la extensión del tumor. La escala oscila entre T1 para tumores pequeños y T4 para los de gran tamaño.
- N: Indica si afecta a ganglios linfáticos. La escala oscila desde N0 para indicar que no hay afectado ningún ganglio pasando por N1 para indicar que están afectados ganglios cercanos al tumor y, por último, N2 y N3 para ganglios más alejados del foco.
- M: Indica si hay metástasis. Esta se define por valores binarios, es decir, M0 indica que no existe metástasis y M1 indican que sí.

#### **2.1.5. Terapias**

Dependiendo del tipo de cáncer, existen diferentes tratamientos, los cuales se pueden usar solos o en combinación. El objetivo de estos es erradicar las células cancerígenas o bien reducir su velocidad de crecimiento.

Los principales tratamientos a día de hoy son:

- Cirugía: Consiste en extirpar todo el tejido tumoral posible.
- Radioterapia: Se trata de una terapia que usa energía ionizante para destruir las células cancerígenas en una región del cuerpo localizada.
- Quimioterapia: Consiste en el empleo de fármacos con el fin de reducir la enfermedad. El principal problema de esta terapia es que dichos fármacos llegan a todos los tejidos del organismo y actúan tanto en las

células malignas como sobre las sanas. Al actuar también los medicamentos sobre estas últimas, da lugar a una serie de efectos secundarios que afecta a la calidad de vida del paciente.

También existen otro tipo de terapias más específicas para determinado tipo de tumores como la inmunoterapia, la hormonoterapia, entre otras [10].

En la actualidad, conocer la evolución de un tumor o predecir el curso que alcanzará ese grupo de células cancerosas es algo complejo de estimar. No obstante, atendiendo a una serie de parámetros específicos, se podría modelar el comportamiento esperable de ese conjunto de células neoplásicas de manera que pudiera ser útil para la caracterización de manera individualizada de cada cáncer. En este sentido, el TFG que se presenta podría ayudar desde un punto de vista clínico a conocer mejor cada tipo de tumor y las respuestas que presentarían en diferentes escenarios temporales.

## 2.2. El juego de la vida

### 2.2.1. ¿Qué es un autómatas celular?

Se trata de un modelo matemático para un sistema dinámico, compuesto por una serie de células o celdas que toman distintos valores o estados. A partir de unas reglas impuestas al modelo, cada celda interactúa con sus celdas vecinas de acuerdo con una función de transición. De esta forma, el autómatas evoluciona en unidades de tiempo discreto [12].

Los elementos básicos de un autómatas celular son los siguientes:

- **Área de estudio:** Es un espacio n-dimensional el cual está homogéneamente dividido en celdas o células. En el caso particular del JLV, se trata de un plano en dos dimensiones (array bidimensional).
- **Conjunto de estados:** Cada célula tiene asignado un valor. En el caso del JLV, existen dos valores posibles para cada casilla. El cero

representa a la célula muerta, mientras que el uno simboliza a la célula viva.

- **Vecindades:** Define el conjunto contiguo de células y la posición relativa de cada una de ellas.
- **Función de transición:** Es la regla de evolución que determina el comportamiento del autómata. Para el JLV, la transición que se utiliza se conoce como vecindad de Moore, donde la célula en estudio depende del estado de sus ocho vecinas más cercanas. (Figura 2.2). De esta forma, las células que componen el autómata cambian sus propios estados de forma simultánea en función de los estados que presentan sus vecinas en un determinado instante de tiempo [13].

		v	v	v		
		v	E	v		
		v	v	v		

**Figura 2.2.** Vecindad de Moore. La celda que contiene la letra “E” representa a la célula en estudio. Las casillas que contienen la “V” simbolizan las células colindantes que deben ser estudiadas (células vecinas).

### 2.2.2. Reglas del Juego de la vida

Las reglas del autómata celular definen a partir del estado actual de cada celda, el próximo estado en pasos de tiempo discretos. Estas reglas se basan en el estado de las células vecinas.

A continuación, se muestran las reglas originales del juego:

- Supervivencia: Si la célula en estudio tiene 2 ó 3 vecinas vivas, esta sobrevive en la siguiente generación.
- Muerte: Si una célula viva tiene menos de 2 células vecinas vivas fallece por soledad en la siguiente generación. En el caso de que una célula tenga más de tres células vecinas vivas, para el siguiente estado, la célula en estudio morirá por sobrepoblación.
- Nacimiento: Si una célula en estudio está muerta y pasa a tener 3 células vivas a su alrededor, en el próximo turno nacerá una nueva célula en dicha posición.

Para facilitar la comprensión del funcionamiento de estas reglas al lector, se expondrá el siguiente ejemplo:

Se tiene la siguiente matriz:

	1	2	3	2	1	
	1	1	2	1	1	
	1	2	3	2	1	

**Figura 2.3.** Ejemplo de array bidimensional en el JLV (Primera generación).

Como podemos observar en la figura 2.3, se representa el plano bidimensional compuesto por células. Las celdas sombreadas representan células vivas y las celdas blancas son células muertas. Los números que se indican en cada celda representan el número de vecinas vivas que cada célula tiene alrededor. Las casillas que no tienen ningún número representado son indicativas de no tener ninguna célula viva colindante. Si aplicamos las reglas originales del JLV redactadas anteriormente, tendremos como resultado la segunda generación de células. Con tan solo mirar la figura 2.3, se puede

determinar las células en la siguiente generación que estarán vivas y cuáles no. Las células que actualmente estén vivas y tengan exactamente dos o tres vecinas vivas permanecerán en el mismo estado para la siguiente generación, mientras que para las células que se encuentran muertas, si no tienen exactamente tres vecinas vivas seguirán muertas. En el caso de que una célula muerta tenga tres vecinas vivas, para la siguiente generación, aparecerá una célula viva en dicha casilla. Por tanto, la evolución de la primera generación a la siguiente sería de la siguiente forma:

		1	1	1		
		2	1	2		
		3	2	3		
		2	1	2		
		1	1	1		

**Figura 2.4.** Representación de la segunda generación de células.

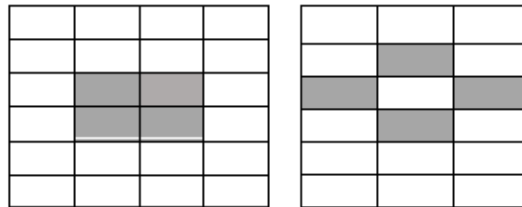
De este modo, para cada iteración se realiza siempre el mismo proceso, es decir, ver el estado de cada celda y en función de sus vecinas, aplicar las reglas correspondientes que determinen la evolución a la siguiente generación. Si se examina la figura 2.4, se podría llegar a la conclusión de que la tercera generación de células quedaría representada de la misma forma que la figura 2.3 [13].

### 2.2.3. Patrones

El JLV puede dar lugar a distintos patrones. No son más que configuraciones específicas de las células que se comportan de una manera determinada. Entre ellos se encuentran:

### 2.2.3.1. Estáticos

Se trata de patrones continuos que debido a su configuración estable permanecen fijos generación tras generación. También son conocidos como patrones inmortales [1].



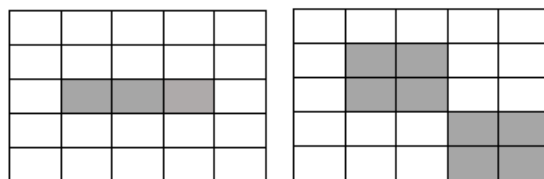
**Figura 2.5.** Ejemplos de patrones estáticos

### 2.2.3.2. Patrones de crecimiento indefinido

Estos patrones se caracterizan por aumentar la población generación tras generación [1].

### 2.2.3.3. Osciladores

Son aquellos que tras evolucionar vuelven a su forma inicial. También se les conoce como periódicos [1].

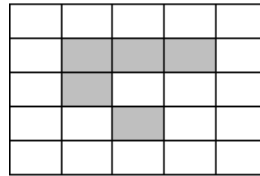


**Figura 2.6.** Ejemplos de patrones osciladores

### 2.2.3.4. Deslizadores

En este caso, del mismo modo que sucede con los osciladores, en el patrón de deslizadores evolucionan hasta volver a llegar a su estado inicial. La

diferencia reside en que estos a su vez se desplazan en el tablero a una velocidad constante. Estos patrones también son conocidos como naves espaciales [1].



**Figura 2.7** Ejemplo de patrón deslizador

## 2.3. Variantes del Juego de la vida

A partir de la versión original, han surgido variantes del mismo. Esto se consigue modificando las reglas que hacen que evolucione este modelo.

Para identificar los diferentes tipos de variantes se utiliza la notación “S/B”:

Se define de la siguiente manera:

- S: Número de células vecinas vivas necesarias para que la célula en estudio sobreviva.
- B: Número de células vecinas necesarias para que la célula en estudio nazca.

Por ejemplo, el JLV se le conoce como “23/3”. El primer par de números simbolizan que una célula seguirá viva si tiene 2 ó 3 células vecinas vivas (en otro caso muere) y en el segundo número representa el número de células vecinas que deben de estar vivas para que se produzca un nacimiento.

Algunos ejemplos de variantes son [13]:



Nombre	S/B	Característica
HighLife	2,3/3,6	Caótico
Long life	5,1/3,4,6	Estable
Day & Night	3,4,6,7,8/3,6,7,8	Estable
Life without death	3/0,1,2,3,4,5,6,7,8	Estable
Morley	2,4,5/3,6,8	Estable
Amebas	1,3,5,8/3,5,7	Caótico

**Tabla 2.1.** Ejemplo de variantes del JLV.

Podemos encontrar el JLV programado en tres dimensiones o incluso se ha llegado a crear una nueva generación del juego, la cual, utiliza funciones continuas y números en coma flotante en lugar de enteros. De esta forma, a la hora de ver el resultado, se obtiene una imagen más suavizada y por tanto, con mayor resolución. A esta implementación se le conoce como SmoothLife [11].



# Capítulo 3: Diseño del programa

## 3.1. Descripción

Se ha implementado un programa basado en el JLV empleando el lenguaje de programación Java y Eclipse como plataforma de desarrollo y creación del mismo.

Se trata de un programa versátil debido a que la ejecución por defecto da lugar al JLV original, sin embargo, se han incluido otras funcionalidades. Simplemente cambiando ciertos parámetros de entrada dará lugar a las distintas versiones del mismo.

Esta herramienta permite:

- Generación de fichero aleatorio de datos.
- A través de la entrada de datos mediante un fichero:
  - Posibilidad de cambiar:
    - Densidad de población inicial de células.
    - Reglas que determinan el comportamiento del autómata celular.
    - Probabilidades que caracterizan el crecimiento celular.
    - El estado inicial en el que se encuentra el tejido a la hora de ser examinado.
    - Determinar a qué velocidad se quiere visualizar la evolución del tejido.
- Guardar y cargar el estado del autómata celular en cada generación para un posible estudio posterior.

La ejecución de la aplicación tiene como resultado un tejido creado por una matriz de células ( $n \times m$ ). El tamaño de esta matriz depende de la entrada de

datos que se le haya proporcionado al sistema, bien sea de forma manual o por generación aleatoria. Si se elige la opción de generación aleatoria, la matriz resultante podrá ser como máximo de 2000 x 2000. (Lo cual equivaldría al estudio de 4000000 células). Se ha elegido esta dimensión por defecto, debido a que, si se pone una mucho más grande, el tiempo de cómputo para el ordenador será mayor. El usuario que desee el estudio de un tejido aleatorio mayor, podrá introducir las dimensiones deseadas mediante consola y el resto de datos se generarán de manera aleatoria. La velocidad a la que se ejecuta la interfaz también está definida por defecto. El usuario que desee una visualización del tejido a mayor velocidad deberá aumentar este valor desde fichero.

A la hora de identificar el tipo de células en la matriz, las casillas blancas representan las células vivas, mientras que las celdas negras simbolizan a las muertas.

## **3.2. Implementación del programa**

A continuación, se van a enumerar y explicar los elementos que componen el programa. A modo de introducción, se representa el diagrama de clases empleado en el programa, con sus correspondientes atributos y métodos en la figura 3.1.

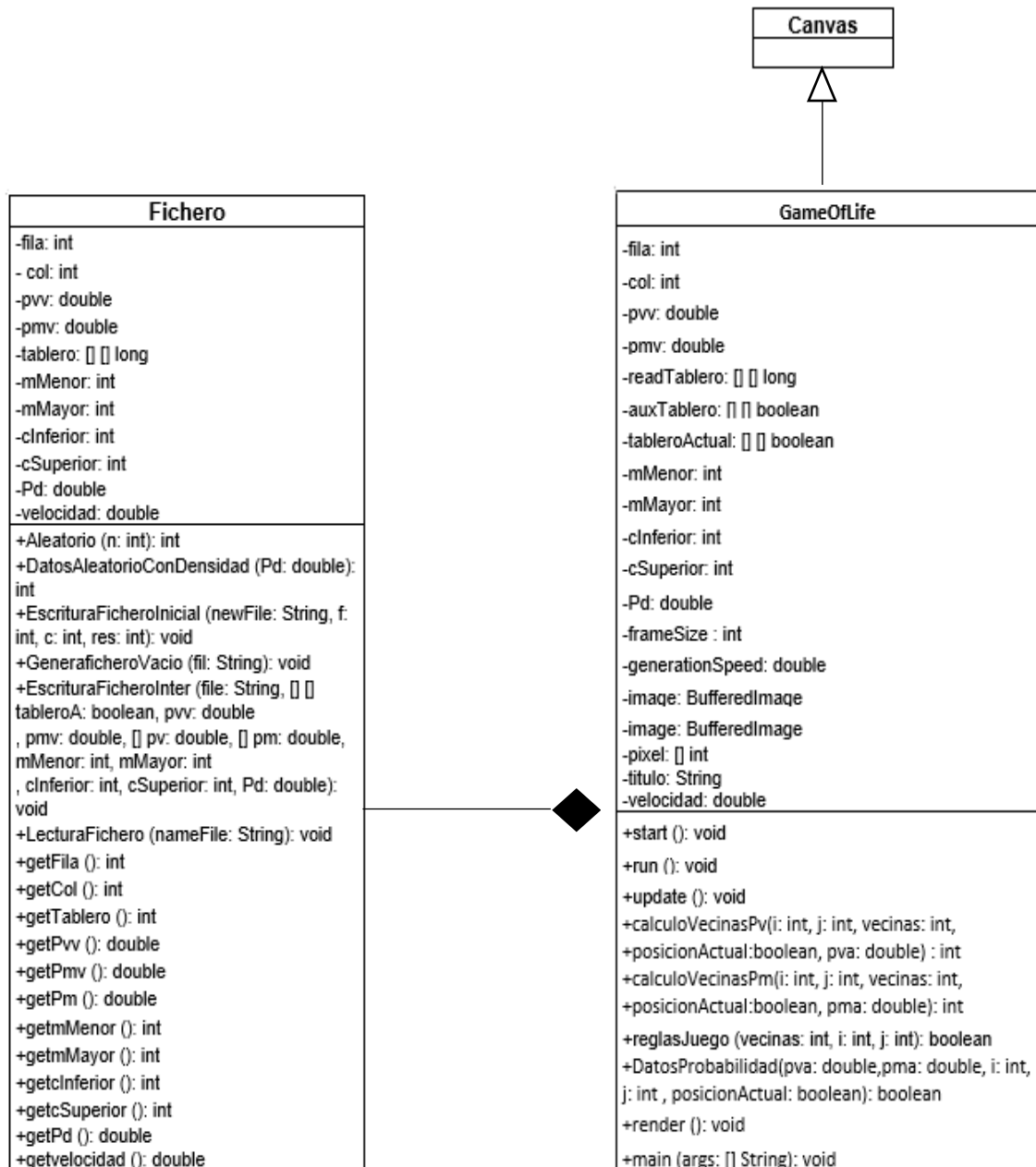
### **3.2.1. Clases**

El proyecto “JuegoDeLaVida” está compuesto por las siguientes clases:

- Clase Fichero: Contiene los métodos necesarios para la escritura, generación y lectura de ficheros de texto. Cuando ejecutamos el programa, la consola pide que se introduzca el nombre del fichero que se desea estudiar. En el caso de que el nombre introducido exista en el directorio seleccionado, el programa leerá ese fichero recogiendo los datos necesarios para la ejecución del programa, de lo contrario, creará y escribirá un nuevo fichero con datos aleatorios o con valores por defecto en función de la elección por parte del

usuario.

- Clase GameOfLife: Se trata de la clase principal del programa. Contiene los métodos que determinan cómo interactúan las células entre sí, además de generar la ventana donde se podrá visualizar el resultado de la ejecución. Para esto último la clase GameOfLife implementa a la clase Canvas perteneciente al paquete Java.awt.



**Figura 3.1.** Diagrama de clases UML.

### 3.2.2. Atributos

Es importante definir algunos atributos de estas clases para la correcta comprensión del programa.

Atributo	Tipo de dato	Descripción
readTablero	Array bidimensional de enteros tipo long.	Recoge del fichero de entrada la matriz inicial que representa al tejido que se quiere estudiar.
tableroActual	Array bidimensional de tipo boolean.	En primer lugar, en ella se vuelca la información contenida en readTablero haciendo un cambio de tipo de datos. Este es el atributo más importante debido a que contiene, para cada generación, el estado de la matriz de células.
auxTableo	Array bidimensional de tipo boolean.	Se trata de una copia de tableroActual. Sobre ella se realiza el estudio y conteo de las células vecinas vivas para cada iteración, sin actualizar su estado hasta que no se han estudiado todas y cada una de las células que contiene la matriz. Mientras tanto, tableroActual en cada iteración se va actualizando. Si no se tuviera esta variable auxiliar, no se realizarían bien los cálculos debido a que el estado de las células cambiaría y daría lugar a errores en la evolución del autómatas.
mMenor	Entero tipo int.	Límite inferior que indica el número de células vecinas que deben de estar vivas para que la célula en estudio estando viva sobreviva a la siguiente generación. Valor por defecto = 2. (simula el JLV original).

Atributo	Tipo de dato	Descripción
mMayor	Entero tipo int.	Límite superior que muestra el número de células vecinas que deben de estar vivas como máximo para que la célula en estudio estando viva no muera en la siguiente generación por sobrepoblación. Valor por defecto = 3 (simula el JLV original).
cInferior	Entero tipo int.	Límite inferior que apunta el número de células vecinas que deben de estar vivas como mínimo para que la célula en estudio estando muerta nazca en la siguiente generación. Valor por defecto = 3 (simula el JLV original)
cSuperior	Entero tipo int.	Límite superior que indica el número de células vecinas que deben de estar vivas como máximo para que la célula en estudio estando muerta nazca en la siguiente generación. Valor por defecto = 3 (simula el JLV original).
Pd	Real en punto flotante tipo double.	Porcentaje de densidad de células vivas inicial. Se utiliza para general el tejido aleatorio inicial. Valor por defecto = 0.35
velocidad	Real en punto flotante tipo double.	Con este atributo se determina la velocidad a la que se ejecuta la interfaz gráfica. Tiene un valor por defecto = 1.0. El usuario incrementado este valor puede obtener una mayor velocidad de visualización
pvv	Real en punto flotante tipo double.	Probabilidad de que estando viva la célula en estudio para la siguiente generación, realmente lo esté. Valor por defecto = 1,0. (JLV original).

Atributo	Tipo de dato	Descripción
pmv	Real en punto flotante tipo double.	Probabilidad de que estando muerta la célula en estudio finalmente nazca. Sería similar a una generación espontánea de la célula. Valor por defecto = 0,0 (Simula JLV original).
pv	Array de reales en punto flotante tipo double.	Probabilidad que se le asigna a cada vecina de la célula en estudio viva. La probabilidad determina si se cuenta como vecina viva o no. Es similar a la introducción de un error a la hora del conteo de células para posteriormente aplicar las reglas que determinan el comportamiento del autómata. En el caso del JLV original $pv = 1,0$ .
pm	Array de reales en punto flotante tipo double.	Probabilidad de que una célula vecina estando muerta se cuente como viva a la hora de conteo de células. Al igual que en la definición anterior, sería equivalente a introducir un error en el momento de conteo de células. En el caso del JLV original $pm = 0,0$ .

**Tabla 3.1.** Definición de los atributos más relevantes del programa

### 3.2.2.1. Descripción de las probabilidades pv y pm.

Como se ha explicado a lo largo de la memoria, el programa va examinando todas las células de la matriz. De forma sucesiva, se detiene en cada casilla analizando el estado de sus células vecinas, realizando un recuento de las vivas y aplicando las reglas que determinan la evolución del autómata para la siguiente generación. Esto corresponde a lo que sucede en el caso del JLV original.

Imagine que la célula en estudio, en este caso, es la celda de color amarillo (i, j) de la figura 3.2. Todas las células colindantes coloreadas de verde corresponderían a sus células vecinas.



		<b>P1</b> <b>(i-1, j-1)</b>	<b>P2</b> <b>(i-1, j)</b>	<b>P3</b> <b>(i-1, j+1)</b>		
		<b>P4</b> <b>(i, j-1)</b>	<b>(i, j)</b>	<b>P5</b> <b>(i, j+1)</b>		
		<b>P6</b> <b>(i+1, j-1)</b>	<b>P7</b> <b>(i+1, j)</b>	<b>P8</b> <b>(i+1, j+1)</b>		

**Figura 3.2.** Ejemplo de matriz de células en estudio.

Como se puede observar en la figura 3.2, a cada vecina se le ha asignado una probabilidad acompañada de un número. Dependiendo del estado de la célula vecina existen dos tipos de probabilidades, las cuales han sido introducidas en la tabla 2:

- En el caso de que la célula vecina se encuentre viva, se le asignará una de las ocho probabilidades contenidas en el array pv. Como se ha explicado en la tabla, esta es la probabilidad de que una célula estando viva se considere como viva a la hora del recuento de células.
- En el caso de que la célula vecina se encuentre muerta, se le asignaría una de las ocho probabilidades contenidas en pm, ya que esta es la probabilidad de que una célula hallándose muerta, se cuente como viva al realizar el cálculo de células vecinas.

Estas dos probabilidades tienen como finalidad la simulación del crecimiento celular en diferentes direcciones del plano, en función de los datos que se definan en estas probabilidades. Dicho de otra forma, puede que exista por parte de las células una prioridad a crecer en una determinada dirección del

plano. Esto se maneja mediante la definición de estas probabilidades. Si se le asigna una mayor o menor probabilidad a una determinada célula vecina se podrá simular diferentes comportamientos, como podría ser una tendencia al crecimiento de forma diagonal por parte del tejido o que las células tiendan a crecer más por un lado que por otro. En el capítulo 4, se mostrará algunos ejemplos de este comportamiento.

### **3.2.2.2. ¿Cómo se asignan las probabilidades pv y pm?**

A cada célula vecina se le asigna una probabilidad como se ha explicado en el apartado 3.2.1.1. Los arrays de probabilidades tienen el siguiente formato:

$$pv = \{pv1, pv2, pv3, pv4, pv5, pv6, pv7, pv8\}$$
$$pm = \{pm1, pm2, pm3, pm4, pm5, pm6, pm7, pm8\}$$

Si se observa la figura 3.2, se puede ver que encima de las posiciones correspondientes a cada celda, hay un número que identifica a cada una de ellas. Por lo tanto, la asignación de las probabilidades viene determinada por ese número. El número de la casilla debe de coincidir con el número de la posición del array correspondiente. De tal forma, el programa identifica el estado de la célula, para saber cuál de las dos probabilidades escoger y en función de la posición de la célula vecina, le asigna una de las probabilidades. Por ejemplo, para la casilla 1, que corresponde a la posición (i-1, j-1), la probabilidad que se le asignaría a esta sería pv1 o pm1 dependiendo de su estado. Expresado de forma genérica, a la casilla N le correspondería, o bien la probabilidad pvN o la pmN en función de su estado.

### **3.2.2.3. Descripción probabilidades pvv y pmv**

Una vez calculado el número de células vecinas vivas que tiene alrededor la célula en estudio, se aplican las reglas que determinan el estado de la célula para la siguiente generación. Existen dos resultados posibles:

1. La célula en estudio sobrevive o nace. En este caso, se aplica la probabilidad pvv, que como se ha descrito anteriormente en la tabla 3.1, es la probabilidad de que habiéndose calculado como viva la

célula en estudio para la siguiente generación, realmente se cuente como viva. Si el valor asignado a esta probabilidad es igual a uno, significa que no existe error, y por consiguiente seguirá viva para la siguiente generación. Conforme este valor disminuye, hay más probabilidades de que la célula aun habiéndose calculado como viva, se considere muerta. Simularía una muerte súbita por parte de la célula.

2. La célula en estudio muere o permanece muerta. En este caso, se aplicaría la probabilidad pmv. Esta es la probabilidad de que habiéndose calculado como muerta la célula en estudio finalmente nazca. Sería similar a una generación espontánea de la célula. Si pmv es igual a cero, significa que no surgirán nacimientos espontáneos por parte de la célula. A medida que se incrementa este valor, habrá menos células muertas en el tejido.

### 3.2.3. Métodos

En siguiente lugar, se van a describir los métodos principales de cada clase, debido a que se trata del pilar fundamental del programa, ya que, constituyen la lógica de la clase o, dicho de otra forma, manipulan el estado del objeto:

#### **Clase Fichero:**

- **DatosAleatorioConDensidad (double Pd):** Esté método sirve para generar la matriz de células inicial de manera aleatoria introduciendo el porcentaje de densidad de población de células inicial. Este devuelve un entero 0 ó 1 (muerto o vivo) en función de si el número aleatorio real generado es mayor o menor respectivamente a esta densidad.
- **EscrituraFicheroInicial (String newFile, int f, int v, int res):** Genera el fichero inicial de datos. El porcentaje de densidad de población inicial y los valores que determinan las reglas de comportamiento del autómata, van a ser siempre por defecto los mismos datos que los empleados en el JLV original. La velocidad a la que se ejecuta la interfaz gráfica también

se define por defecto. El usuario puede determinar, a través de consola, el tamaño de tejido que desea estudiar o bien indicar que las dimensiones sean aleatorias, además de indicar si los datos de probabilidad los desea por defecto (simulando el JLV) o aleatorios. La velocidad de visualización de la interfaz se podrá cambiar a través de fichero. La matriz de células se generará de manera aleatoria según los datos introducidos.

- **GeneraficheroVacio (String fil):** Este método genera un fichero de texto donde solo se especifica que datos hay que introducir en cada lugar (Ver figura A.6). Se utiliza en el caso de que el usuario quiera introducir manualmente todos los datos.
- **LecturaFichero (String nameFile):** Este método lee los datos recogidos en el fichero de texto y los guarda en variables para su posterior uso en el programa.
- **EscrituraFicheroInter (String file, Boolean [] [] tableroA, double pvv, double pmv, double [ ] pv, double [ ] pm, int mMenor, int mMayor, int Inferior, int cSuperior, double Pd ):** Este método es empleado para escribir los ficheros de texto que guardan el estado de cada una de las generaciones del autómata. Escribe el estado de la nueva matriz de células y reescribe el resto de datos, los cuales, permanecen igual para todas las generaciones, a excepción del porcentaje de células vivas que también se actualiza para cada generación.

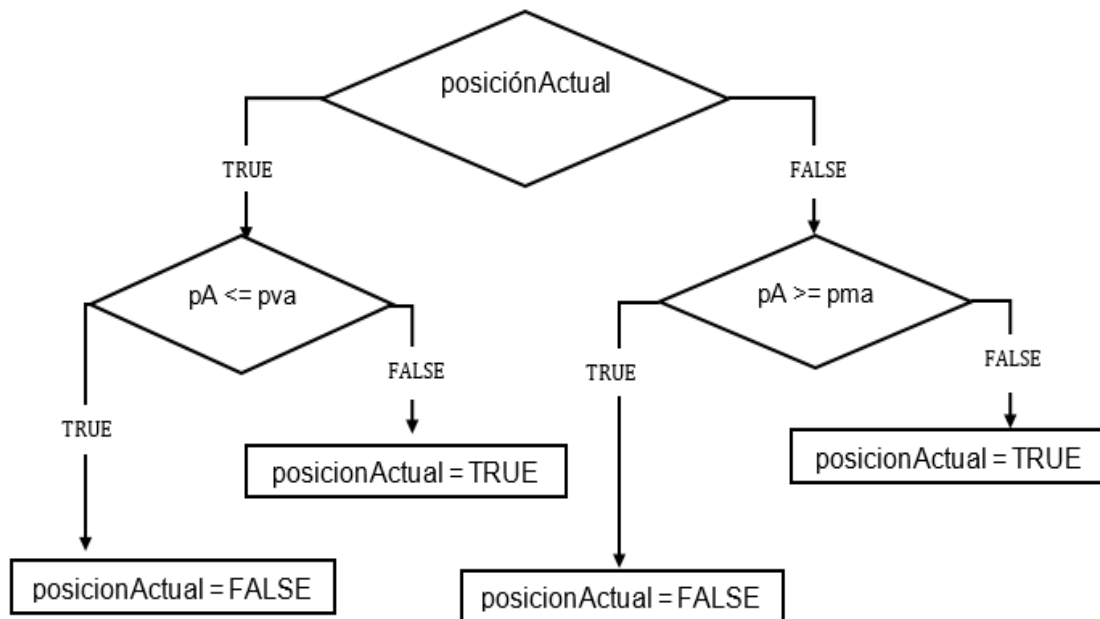
#### **Clase GameOfLife:**

- **start ():** Este es el primer método que se ejecuta en la clase GameOfLife. Crea tableroActual y auxTablero. A partir de la matriz de células leída desde fichero y guardada en readTablero, se hace una conversión de tipo de enteros a booleanos y se guarda en el variable tableroActual.
- **run ():** Este realiza un bucle donde se llama al resto de métodos de la clase GameOfLife de manera iterativa hasta el cierre del programa. En

este método también se define la velocidad a la que se verá la evolución del tejido.

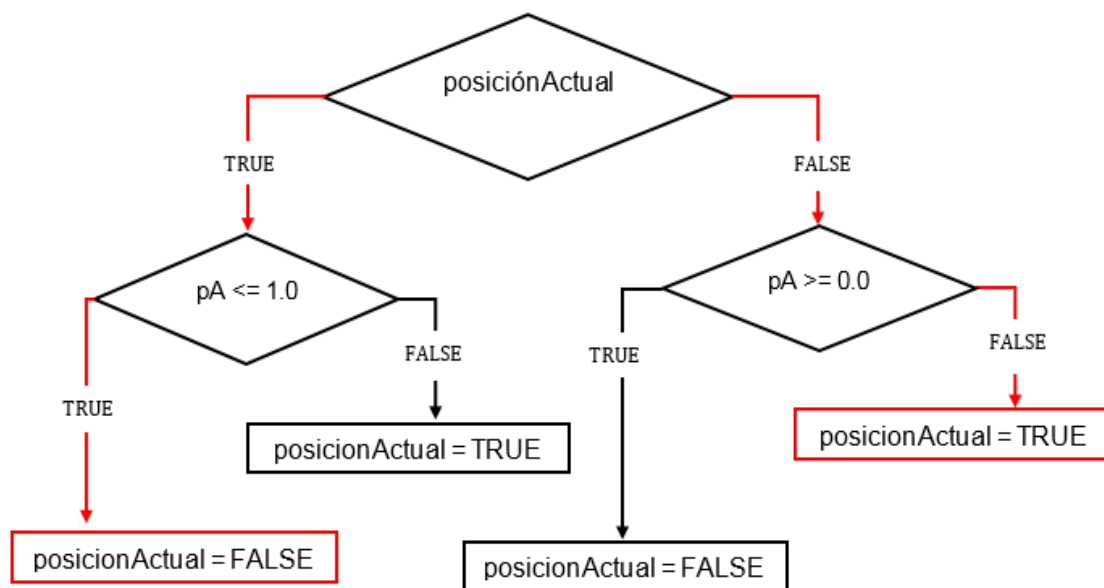
- **update ():** Este es invocado por el método run (). Mediante un bucle anidado, recorre la matriz de células estudiando cada una de sus posiciones. Para cada celda, se determina el número de células vivas colindantes con la ayuda de los métodos calculoVecinasPv, en el caso de que en la célula vecina en estudio se encuentre viva o en caso contrario, con calculoVecinasPm. Una vez se tenga el recuento de todas las células vivas alrededor de la célula en estudio, se dispondrá a llamar al método reglasJuego, el cual aplicará las reglas que determinan el estado del autómatas para la siguiente generación. Este proceso se repite sucesivamente con cada una de las casillas de la matriz, en cada una de las generaciones.
- **DatosProbabilidad (double pva, double pma, int i, int j, boolean posicionActual):** Este método es invocado por calculoVecinasPm, calculoVecinasPv y reglasJuego, los cuales se describirán más adelante. Su finalidad es estudiar la probabilidad que tiene la célula en estudio de estar realmente viva o muerta en función del estado en el que se encuentre en ese momento. Este método genera una probabilidad aleatoria (pA) entre 0.0 y 1.0, la cual, compara con las probabilidades recogidas inicialmente del fichero y en función del resultado obtenido, determina el estado final de cada casilla. La variable pva representa la probabilidad de que una célula estando viva permanezca viva mientras que, la variable pma es la probabilidad de que una célula muerta finalmente se considere viva. Por esta razón, a este método se le pasa como parámetro una variable que representa la posición actual de la célula en estudio (posicionActual). En el caso de que la casilla que se haya pasado como posición actual se encuentre viva, se comparará la pA generada con pva. Si pva es menor o igual a la pA, querrá decir que la célula se considerará viva. En caso contrario, se estimará como muerta. Si en lugar de eso, en la posición actual se encuentra una célula

muerta, la  $pA$  se comparará con  $pma$ . En la situación de que la  $pA$  sea mayor o igual que  $pma$ , la célula permanecerá en su estado, es decir, muerta. En caso contrario, se considerará viva. Finalmente, el método devuelve un boolean con el estado final de la posición actual. A continuación, en la figura 3.4 se muestra un diagrama de flujo de este método.



**Figura 3.4.** Diagrama de flujo genérico del método *DatosProbabilidad*.

A como de ejemplo, si los datos pasados por fichero fueran los del JLV,  $pva$  sería igual a 1.0 mientras que  $pma$  sería igual a 0.0. Por lo que, si se sigue el diagrama de flujo de la figura 3.5, se puede determinar que en este caso no existe ninguna clase de error, por lo tanto, en el caso de que la célula en estudio se encuentre viva, no habrá  $pA$  mayor que  $pva$ , o si bien, la célula de la posición actual se encuentra muerta, no habrá  $pA$  menor que  $pma$  y como resultado, la célula se considerara muerta.



**Figura 3.5.** Diagrama de flujo del método *DatosProbabilidad* para el JLV. Las flechas rojas indican el camino a seguir por el programa.

- **calculoVecinasPv(int i, int j, int vecinas, Boolean posicionActual, double pva):** Se emplea en caso de que la célula vecina en estudio se encuentre viva (la posición de esta corresponde al parámetro `posiciónActual`). Este método invoca a *DatosProbabilidad* (`pva, 0.0, fila, columna, posicionActual`). La variable `pva` es un valor del array `pv` perteneciente al fichero introducido al ejecutar el programa inicialmente (Ver apartado 3.2.2.1). Este valor es determinado por la posición de la célula vecina viva en estudio. La variable `pma` se fija con el valor 0.0, ya que la célula en estudio se encuentra viva. Si *DatosProbabilidad* devuelve `true`, el método `calculoVecinasPv` cuenta a esta vecina como viva, añadiéndola al contador de vecinas. En caso contrario, no se añadirá a dicho contador. El método devuelve un entero a `update`, para más tarde, una vez estudiadas todas las células del tablero, ser pasado por parámetro a `reglasJuego`.
- **calculoVecinasPm (int i, int j, int vecinas, Boolean posicionActual, double pma):** Se usa al contrario que `calculoVecinasPv`, es decir,

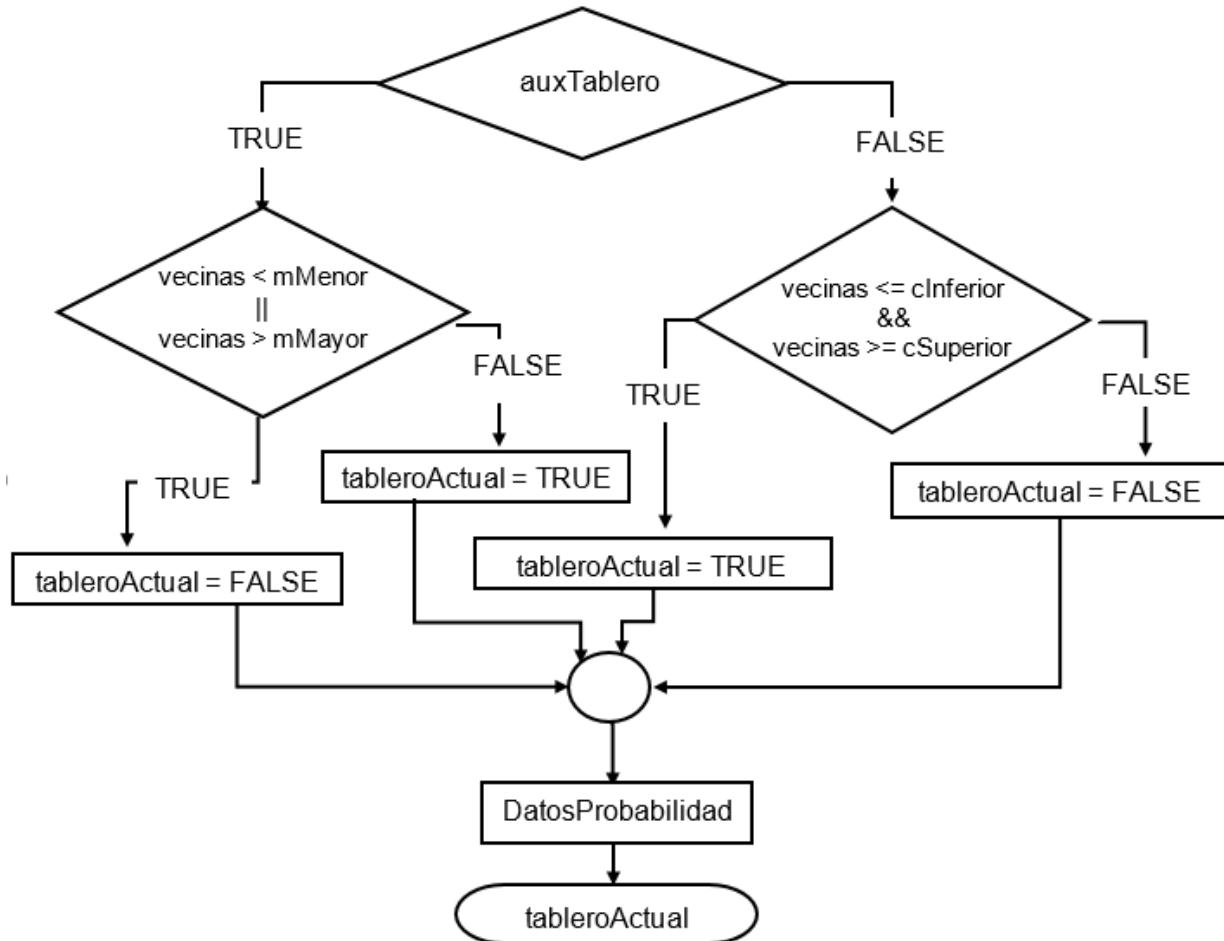
cuando se encuentre muerta la célula vecina en estudio. Llama al método `DatosProbabilidad` (1.0, pma, fila, columna, posicionActual) estableciendo en este que pva sea igual a 1.0, debido a que se trata de una célula muerta y pma corresponde a un valor del array pm del fichero de datos, el cual depende de la posicionActual que se pasa por parámetro y determina su elección (Ver apartado 3.2.2.2). Si `DatosProbabilidad` devuelve true, cuenta a esta vecina como viva añadiéndola al contador de células y al igual que en el método anterior, devuelve un entero a `update` para una vez terminado el recuento de células vecinas vivas, enviar el dato a `reglasJuego` y poder aplicar las reglas impuestas al autómata.

- **`reglasJuego (int vecinas, int i, int j)`**: Este método impone las reglas que harán que el autómata celular evolucione de una manera determinada. El parámetro `vecinas` indica el número de células vecinas vivas que tiene la célula en la posición `i` (fila), `j` (columnas) de la matriz en estudio. Tras aplicar las reglas y tener el resultado del estado de la celda en estudio, este método invoca a `DatosProbabilidad` para determinar mediante las probabilidades pasadas por fichero (pvv, pmv), si esa célula finalmente se queda en el estado calculado por `reglasJuego` o cambia. El resultado se guardará en `tableroActual` y formará parte de la matriz de células para la siguiente generación. En la figura 3.6 podemos observar el diagrama de flujo de este método.

Es importante recalcar que, aunque `DatosProbabilidad` dé como resultado un valor de `posicionActual` distinto al introducido por parámetro, este no cambia la matriz en estudio, es decir, solo devuelve el resultado obtenido al método que lo invoca. Por ejemplo, para `calculoVecinasPv` y `calculoVecinasPm` solo servirá para determinar el número de células que se considerarán vivas finalmente, pero en ningún momento producirá un cambio en la matriz. Esto sería similar a introducir un error en el conteo de células. En el método `reglasJuego`, si se producirá un cambio en `tableroActual` dependiendo del valor obtenido por `DatosProbabilidad`, ya que así el método lo impone. Esto de igual



forma, supondría un error en el caso de que DatosProbabilidad diera un resultado distinto reglasJuego, con la diferencia de que afectaría a la matriz de estudio para la siguiente generación.



**Figura 3.6.** Diagrama de flujo del método reglasJuego.

- **render ():** Genera la ventana donde se visualizan las generaciones del autómata celular. En este método definimos que las celdas que estén de color blanco representarán las células vivas mientras las que estén en negro serán las células muertas.



## Capítulo 4: Pruebas y resultados

En este capítulo se van a realizar simulaciones con el programa introduciendo distintos parámetros para observar cómo se comporta el autómatas celular probabilístico.

Es importante resaltar que la interpretación actual del modelo se encuentra desarrollada en un entorno que podríamos considerar genérico. Cada población celular, cada tejido y cada órgano, tendrán unas peculiaridades específicas que deberían verse reflejadas en un futuro en estas simulaciones. A modo de ejemplo, tenemos células muy sensibles a la falta de oxígeno, como sucede en el cerebro, así como otras células cuya subsistencia depende mucho de la vecindad de otras células, como puede ser en el cartílago.

La supervivencia de determinados grupos celulares dependerá de varios elementos, que podemos esquematizar atendiendo al metabolismo celular en:

1. Aporte apropiado de nutrientes, tales como glucosa u oxígeno, por los vasos sanguíneos, en cantidad suficiente que permitan a las células poder mantener un estado funcional apropiado.
2. Capacidad de eliminar los tóxicos derivados del metabolismo celular a la circulación, para evitar que se acumulen y generen problemas en el tejido.
3. Espacio intersticial proporcionado al número de células, que permita garantizar un paso de nutrientes/tóxicos a los vasos sanguíneos.

Teniendo en cuenta lo anterior, podríamos encontrar determinadas circunstancias, en las que una sobrepoblación celular, pudiera generar la muerte en la siguiente generación por falta de aportes nutricionales o déficit en la eliminación de sustancias tóxicas. Por otro lado, la presencia de células aisladas, hacen más vulnerable al tejido, al tener que difundir con mayor dificultad a través del intersticio las sustancias previamente mencionadas, así como sus propiedades estructurales pueden verse afectadas. A su vez, en este último caso, la capacidad funcional de las células también pudiera verse incrementada en exceso, al intentar suplir la carencia de otras células,

haciendo su metabolismo insuficiente para poder cubrir la demanda esperada. A modo de ejemplo, indicar que determinados tumores con un crecimiento descontrolado, terminan necrosándose en determinados puntos, dado que el crecimiento celular no ha sido acompañado de un aumento de circulación sanguínea a ese nivel. En definitiva, podemos resumir que el exceso o defecto de células, genera situaciones disfuncionales en los tejidos, hecho que compromete la viabilidad de las siguientes generaciones celulares, tal y como refleja el modelo presentado en este TFG.

#### 4.1. Simulación 1: Juego de la vida original

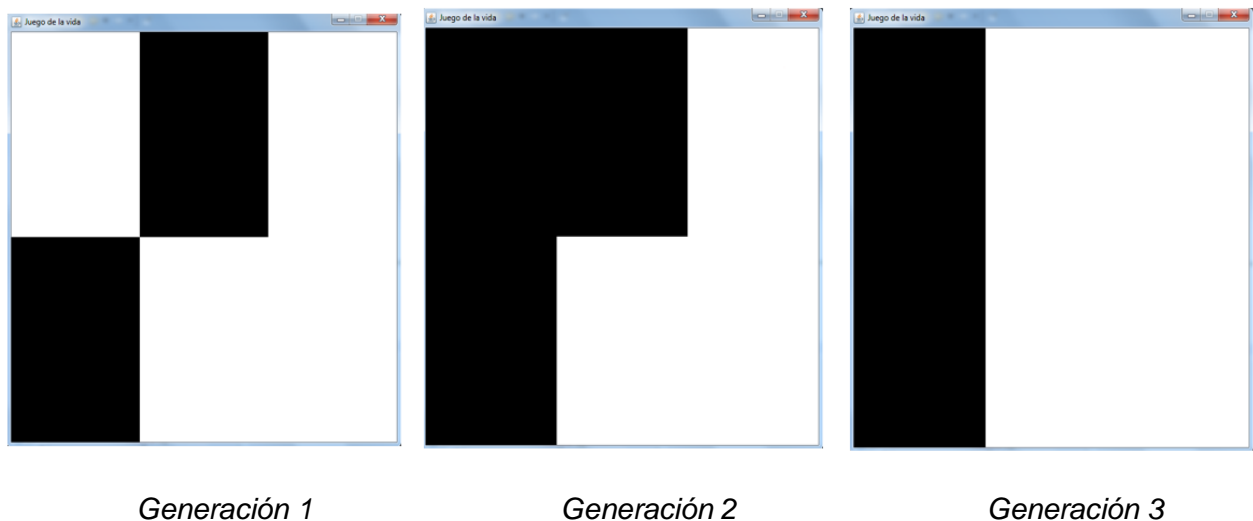
En esta experiencia se va a seguir los patrones del JLV original. Esta simulación se va a realizar con un tejido pequeño para poder comprobar que realmente el autómatas evoluciona de manera correcta.

A partir de la matriz:

1	0	1
0	1	1

**Figura 4.1.** Matriz correspondiente a la simulación 1

Obtenemos el siguiente resultado:



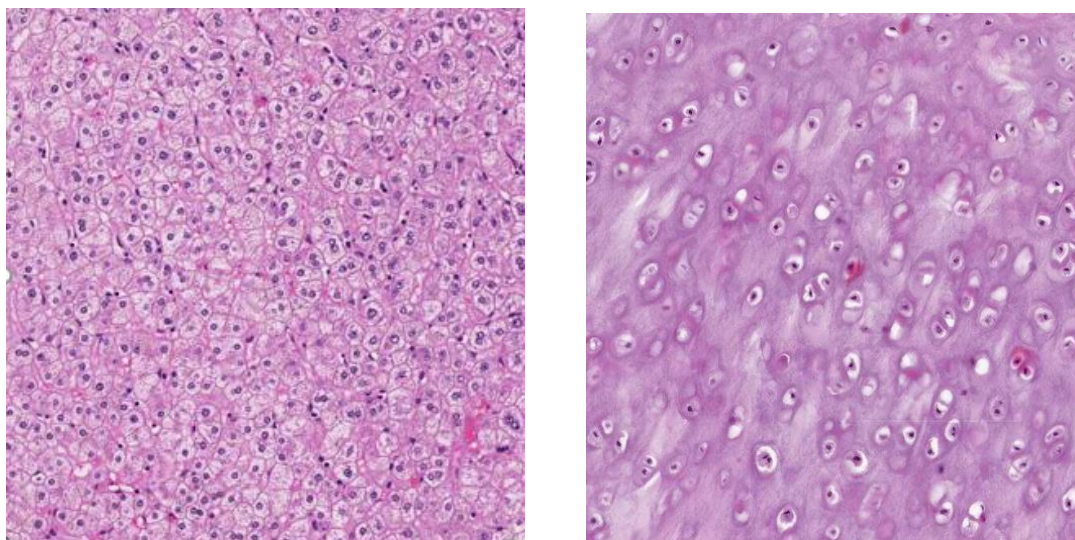
**Figura 4.2.** Simulación de la matriz de la figura 4.1 con el JLV.

En la figura 4.2 se representan las tres primeras generaciones que derivan de la matriz representada en la figura 4.1. A partir de la tercera generación, el autómatas celular permanece estable debido a la configuración que alcanzan sus células (Apartado 2.2.3).

## 4.2. Simulación 2: Variación de densidad de población

El porcentaje de densidad de población celular ( $P_d$ ) hace referencia al número de células que habrá inicialmente en el tejido creado de forma aleatoria. Cuanto más alto sea este porcentaje, mayor será la probabilidad de que haya células vivas.

Es importante destacar que no todos los tejidos y órganos tienen la misma cantidad de células y, por tanto, deberían adaptarse los resultados de este TFG, de manera específica según el modelo que pretendamos generar. Por ejemplo, el porcentaje de células que existe en el tejido hepático, cuya función es depurar sustancias tóxicas y generar determinados factores, es mayor que en el tejido cartilaginoso, cuya función es meramente estructural. (Figura 4.3)



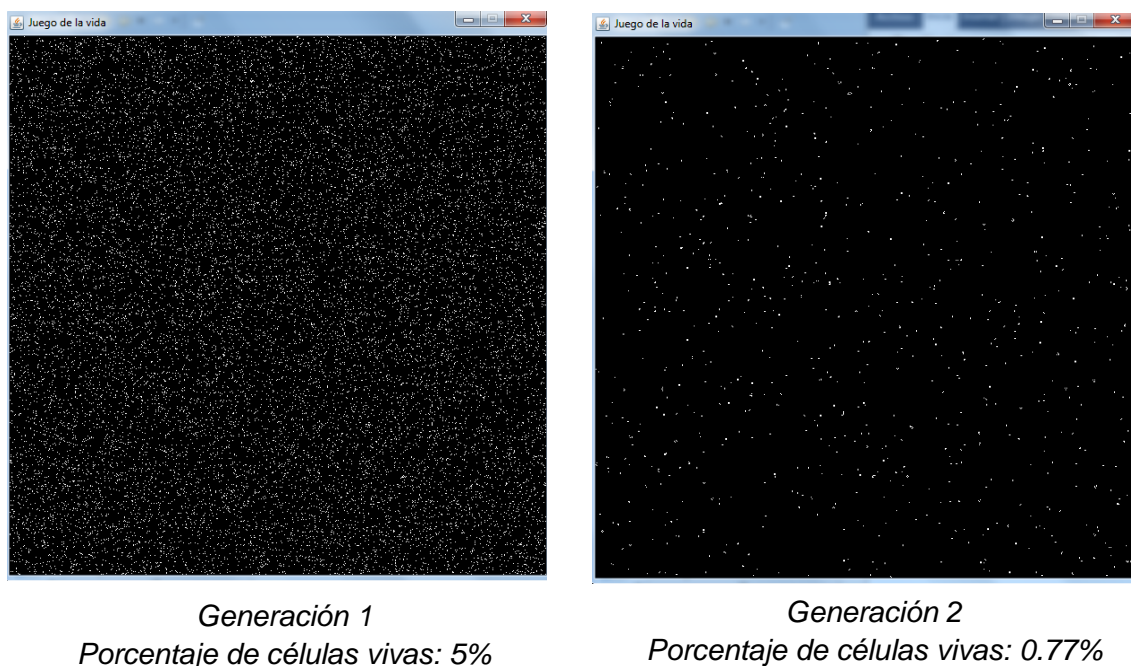
**Figura 4.3.** [15] [16] Diferencia de densidad de células entre tejido hepático (imagen izquierda) y tejido cartilaginoso (imagen derecha)

Por tanto, realmente este valor va a depender del tejido concreto que el usuario quiera estudiar. En este trabajo no se emplea ningún tejido en

concreto, sino que son tejidos generados de forma aleatoria y en condiciones ideales.

A continuación, se exponen algunos ejemplos con distintos parámetros para objetivar de qué forma el Pd afecta al comportamiento celular. En las siguientes experiencias, se usará una matriz de 900 x 900 células la cual, seguirá las mismas probabilidades y reglas que el JLV original.

- Para un  $Pd = 0,05$  (5%) se obtendrán los siguientes resultados:



**Figura 4.4.** Simulación con un porcentaje inicial de densidad de población del 5%

Cuando se realiza la simulación con un Pd pequeño, el resultado es un tejido donde no abundan las células vivas. El porcentaje de células vivas es mayor en la generación 1 que en la 2. Esto se corrobora a simple vista en la figura 4.4, ya que prevalece el color negro frente al blanco, el cual indica mayor presencia de células muertas.

Cuando existe un déficit de células, como sucede en este caso, se puede interpretar que el tejido podría ser disfuncional. En primer lugar, por no disponer de un número de células suficientes para alcanzar un rendimiento metabólico específico apropiado. En segundo lugar, las condiciones

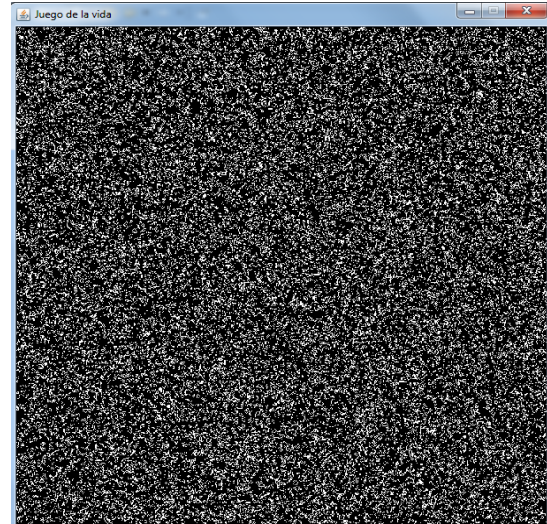


arquitectónicas de la matriz celular/espacio intersticial, es demasiado amplia para permitir un adecuado sostén del tejido y difusión de sustancias. En consecuencia, la evolución probable del tejido será progresivamente a la muerte celular en las siguientes generaciones.

- Para un  $P_d = 0,45$  (45%) se obtendrán los siguientes resultados:



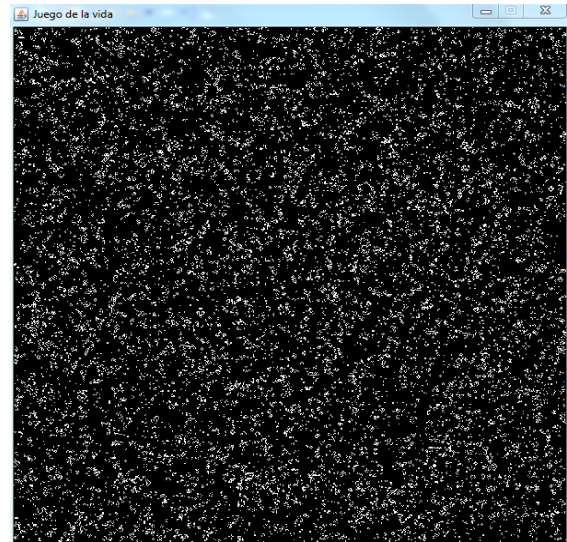
*Generación 1*  
*Porcentaje de células vivas: 45%*



*Generación 12*  
*Porcentaje de células vivas: 20.7%*



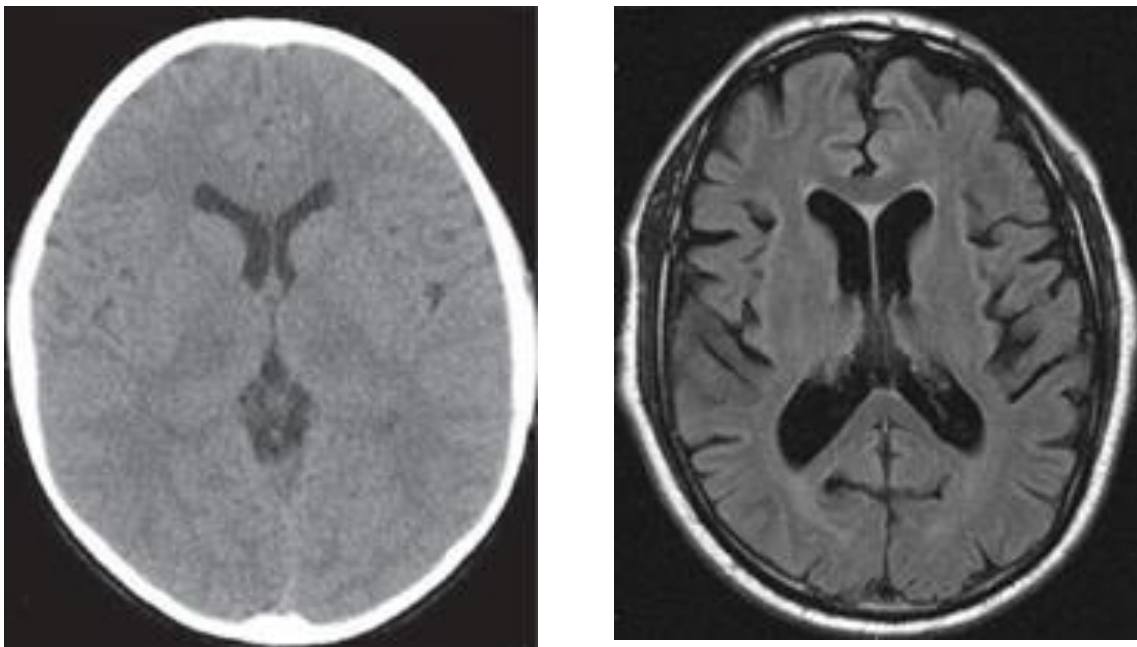
*Generación 27*  
*Porcentaje de células vivas: 15.6 %*



*Generación 115*  
*Porcentaje de células vivas: 8.9 %*

**Figura 4.5.** Porcentaje de densidad inicial de población del 45%

En este caso, la simulación se ha realizado con un Pd del 45%. Como se puede observar en las cuatro imágenes de las sucesivas generaciones (desde la generación 1 hasta la generación 115) de la figura 4.5, existe en los últimos casos una tendencia a estabilizarse la relación visual de células vivas y muertas con el paso de tiempo, aunque de manera progresiva el tejido terminaría desapareciendo. Este patrón podría ser parecido al proceso de senescencia natural que sucede en nuestro organismo. Conforme se suceden generaciones celulares se van generando procesos que disminuyen el número y por tanto la funcionalidad de los tejidos. A modo de ejemplo, la densidad celular del cerebro va disminuyendo de manera progresiva en el ser humano, de hecho la demencia senil es una consecuencia habitual de estos hechos. Inclusive mediante pruebas de imagen se puede valorar como la cantidad de tejido cerebral progresivamente va reduciéndose a lo largo de los años (ver figura 4.6).

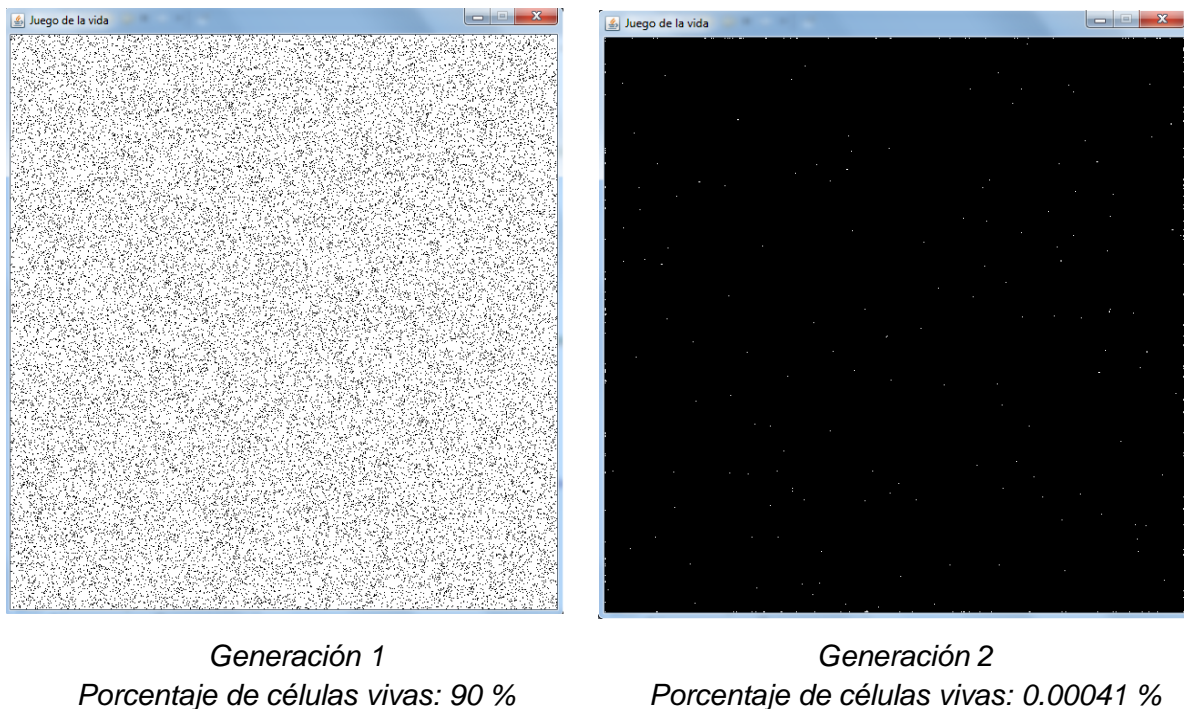


**Figura 4.6.** Diferencia entre tomografía axial computarizada cerebral normal (imagen izquierda) y una con reducción de masa cerebral correspondiente a demencia senil (imagen derecha)



- Para un  $P_d = 0.9$  (90%) se obtendrán los siguientes resultados:

Al emplear un  $P_d$  igual al 90%, se crea una matriz con una gran cantidad de células vivas. Esto se puede ver a simple vista en la generación 1 de la figura 4.7, ya que en la representación gráfica predominan las células de color blanco. Esto se debe a que en la primera generación existe una sobrepoblación de células vivas lo cual conlleva a la muerte de las mismas para la segunda generación.



**Figura 4.7.** Generación de tejido con una densidad de población del 90%

Este comportamiento se debe a una sobrepoblación del tejido. El exceso de células puede conllevar a estas a no recibir el aporte de oxígeno suficiente o no poder eliminar residuos de manera apropiada como se comentó al inicio del capítulo.

Los tumores suelen tener un comportamiento similar a este, puesto que tienden a crecer de manera descontrolada, hasta tal punto, que generan masas de células que, al no poder ser irrigadas por la sangre de una manera apropiada, finalmente terminan necrosándose. En la figura 4.8 se aprecia una

pieza de cáncer de pulmón con áreas necróticas (zonas de color negro). Esto representa la muerte de las células del propio tumor.



**Figura 4.8.** [17] *Pieza de tumor pulmonar con áreas necróticas.*

### **4.3. Simulación 3: Variación de reglas.**

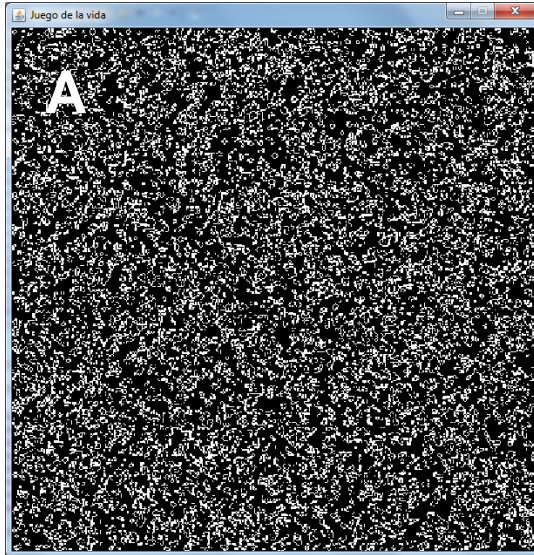
Para esta simulación, se van a modificar las reglas que determinan el comportamiento del autómata celular. Las probabilidades usadas en este apartado son las mismas que en el JLV original.

Se van a presentar cuatro simulaciones para un mismo tejido, cuyas dimensiones son de 400 x 400 células. El objetivo es visualizar el desarrollo celular, para un mismo tejido, con la imposición de reglas diferentes.

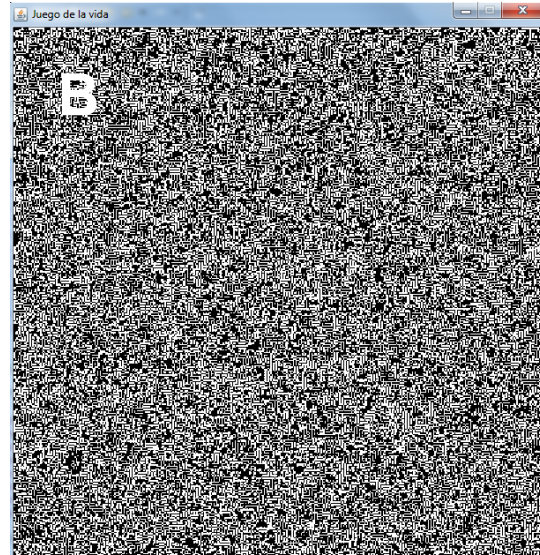
La figura 4.9 está compuesta por cuatro simulaciones:

1. La simulación superior izquierda (A) corresponde a un tejido tratado con las reglas del JLV original.
2. En la simulación superior derecha (B), se ha modificado del fichero de entrada el parámetro *mMayor*. Tal y como se cita en el apartado 3.2.2, *mMayor* es el límite superior que marca el número de células vecinas que deben estar vivas como máximo, para que la célula en estudio no muera en la siguiente generación por sobrepoblación. En este caso, el valor de *mMayor* será cuatro.

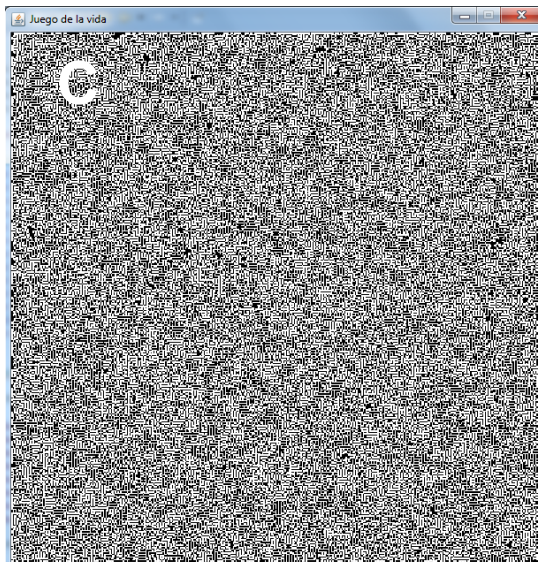
3. En la simulación inferior izquierda (C), se ha aumentado de nuevo el parámetro  $m_{\text{Mayor}}$ , siendo en esta ocasión igual a cinco.
4. Por último, en la simulación inferior derecha (D), se ha establecido el valor máximo de  $m_{\text{Mayor}}$ , y se ha incrementado el intervalo de valores de  $c_{\text{Inferior}}$  y  $c_{\text{Superior}}$  (ver Tabla 4.1).



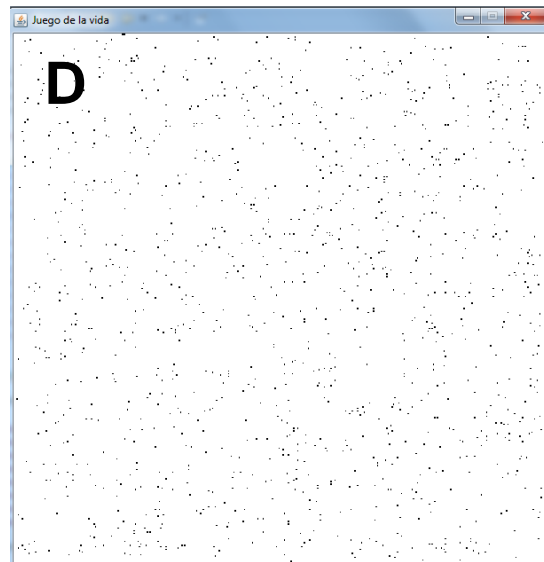
*Generación 8*  
*Porcentaje de células vivas: 24.1 %*



*Generación 8*  
*Porcentaje de células vivas: 45.5 %*



*Generación 8*  
*Porcentaje de células vivas: 55.8 %*



*Generación 8*  
*Porcentaje de células vivas: 99.2 %*

**Figura 4.9.** Representación de la generación 8 de un mismo tejido con diferentes reglas que rigen el comportamiento celular.

Parámetro	Valor
mMenor	2
mMayor	8
cInferior	3
cSuperior	7

**Tabla 4.1.** *Parámetros empleados en la simulación D de la figura 4.9.*

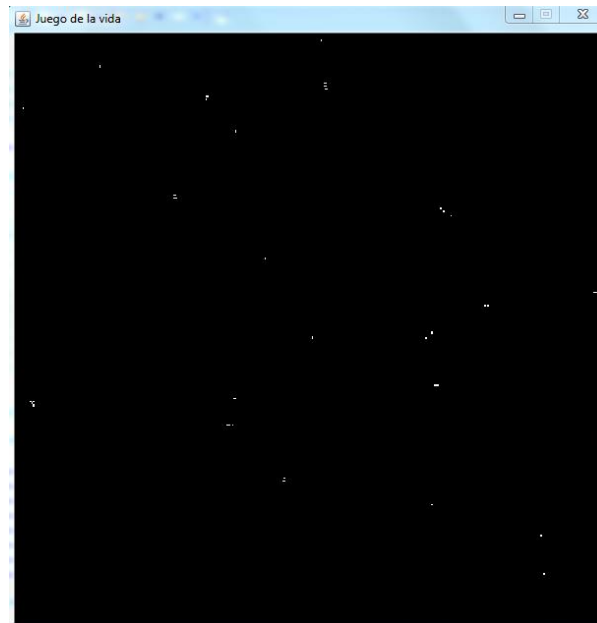
La conclusión que se puede obtener al observar las cuatro simulaciones es el aumento del porcentaje de células vivas en cada una de ellas. Esto se debe, en la simulación B y C, al incremento del parámetro mMayor. En la cuarta simulación además se ha ampliado el rango que debe cumplirse para la génesis de una nueva célula. Esto conlleva a un mayor crecimiento celular en comparación con el resto de simulaciones (Descripción detalla en Anexo A).

En conclusión, si se varían los valores que determinan las reglas, se puede conseguir que las células del tejido se comporten de una determinada manera. Si se amplían los valores de los parámetros que determinan las reglas, se conseguirá un predominio de células vivas en la matriz, las cuales serán más resistentes y permanecerán más tiempo en el tejido.

La figura 4.10 presenta una situación diferente, al imponerse reglas restrictivas. Por tanto, será más difícil para las células sobrevivir o expandirse de manera descontrolada. Para estudiar este escenario se ha empleado el mismo tejido que en las cuatro simulaciones anteriores y se ha determinado un rango más alto donde las células vivirían. (Ver Tabla 4.2)

Parámetro	Valor
mMenor	6
mMayor	8
cInferior	4
cSuperior	5

**Tabla 4.2.** *Parámetros empleados para la simulación de la figura 4.10.*



*Generación 5  
Porcentaje de células vivas: 0.00034 %*

**Figura 4.10:** Simulación de la generación 5 del tejido empleado en la figura 4.9 con diferentes reglas que rigen el comportamiento celular.

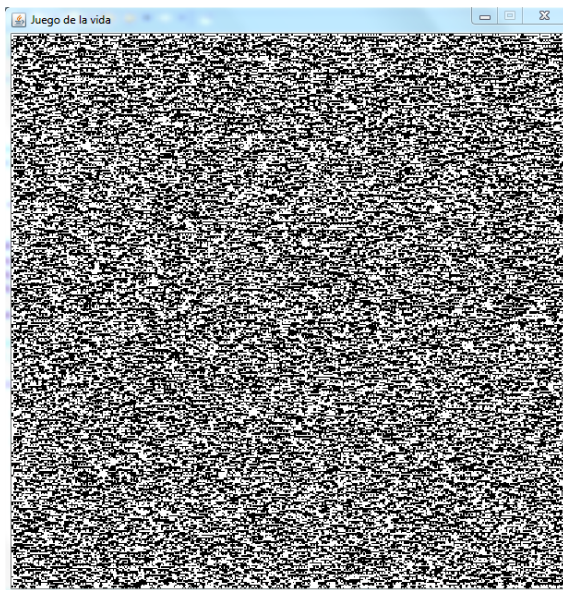
De esta manera, si se hace un estudio posterior de este TFG, se podría buscar la correlación que existe entre un tejido específico y las reglas que rigen el comportamiento del autómata en situaciones normales como patológicas (isquemia del tejido, activación tumoral, intoxicación por sustancias, etc).

## **4.4. Simulación 4: Variación de reglas y probabilidades.**

### **4.4.1. Variación de reglas y probabilidades pv y pm**

En la siguiente simulación, se mostrará un tejido que comienza un proceso de crecimiento descontrolado. Para ello, se va a emplear una matriz cuyas dimensiones son de 400 x 400 células.





*Generación 1*  
*Porcentaje de células vivas: 50 %*



*Generación 130*  
*Porcentaje de células vivas: 61 %*



*Generación 210*  
*Porcentaje de células vivas: 63 %*



*Generación 715*  
*Porcentaje de células vivas: 78.3 %*

**Figura 4.11.:** *Simulación de tejido con crecimiento descontrolado.*

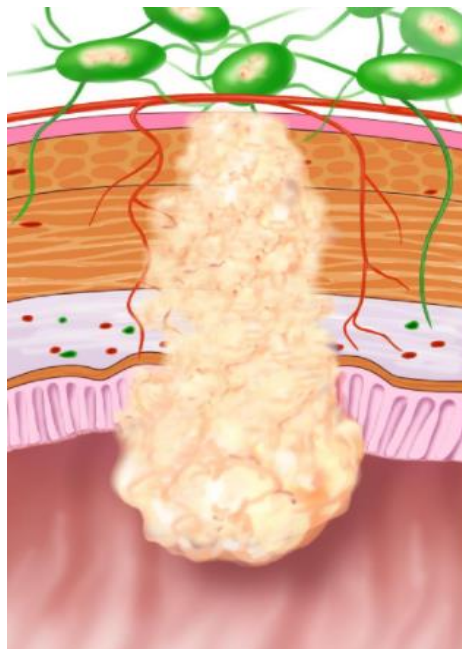
Para alcanzar esta configuración se han modificado las variable pv y pm. De esta manera, se genera un crecimiento con un patrón progresivo de carácter ascendente. Conforme van transcurriendo las generaciones, mayor es el porcentaje de células vivas. Los parámetros modificados para representar la

simulación son los recogidos en la Tabla 4.3. Los valores que no se recogen en la tabla son los mismos empleados en el JLV original.

Parámetro	Valor
mMenor	1
mMayor	5
clInferior	3
cSuperior	3
pv	{1.0, 1.0, 1.0, 0.0001, 0.0001, 0.0, 0.0, 0.0}
pm	{0.001, 0.001, 0.001, 0.98, 0.98, 0.98, 0.98, 0.98}

**Tabla 4.3.** *Parámetros empleados para la simulación de la figura 4.11.*

Estos cambios pueden ser los que se objetivan en determinados tejidos en los que se activan procesos oncológicos proliferativos, que llegan inclusive a invadir tanto la matriz propia del tejido como la de otros tejidos y órganos subyacentes. En estos casos se habla del carácter infiltrante del tumor, dado que no respeta estructuras anatómicas y espacios entre diferentes poblaciones celulares.

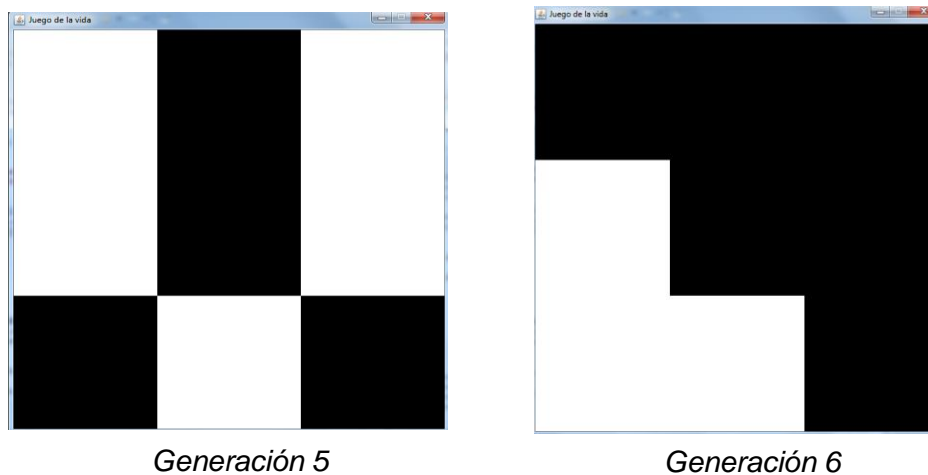


**Figura 4.12.** [18] *Infiltración de masa tumoral*

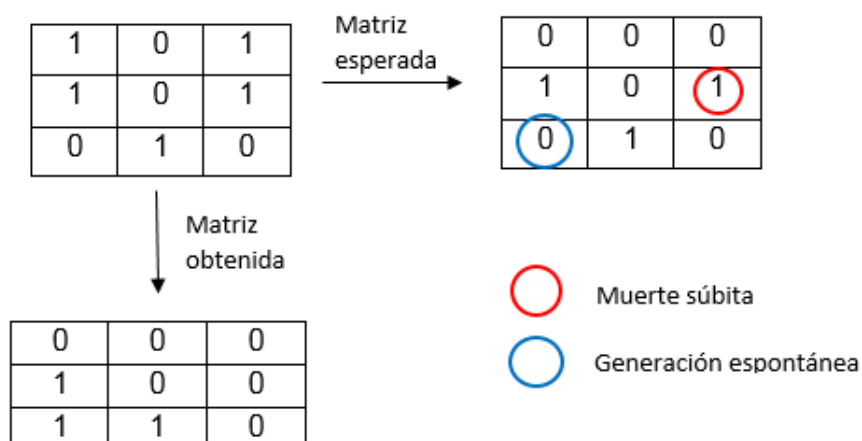
#### 4.4.2. Variación de reglas y probabilidades pvv y pmv

Para poder observar realmente la función de pvv y pvm, se va a emplear un tejido con unas dimensiones 3x3. Por tanto, utilizaremos una matriz pequeña siguiendo las mismas reglas y probabilidades pv y pm que las descritas en el JLV. Se establecerán un pvv de 0,89 y un pvm de 0,25. De esta forma, a menor escala, podemos apreciar de manera simplificada las alteraciones celulares generadas en las sucesivas generaciones.

Si se examina la figura 4.13, se puede llegar a la conclusión de que el programa en este ejemplo, no respeta las reglas del JLV original. Esto se debe a la introducción de las probabilidades pvv y pmv (Figura 4.14).



**Figura 4.13.** Simulación JLV variando los parámetros pvv y pmv.

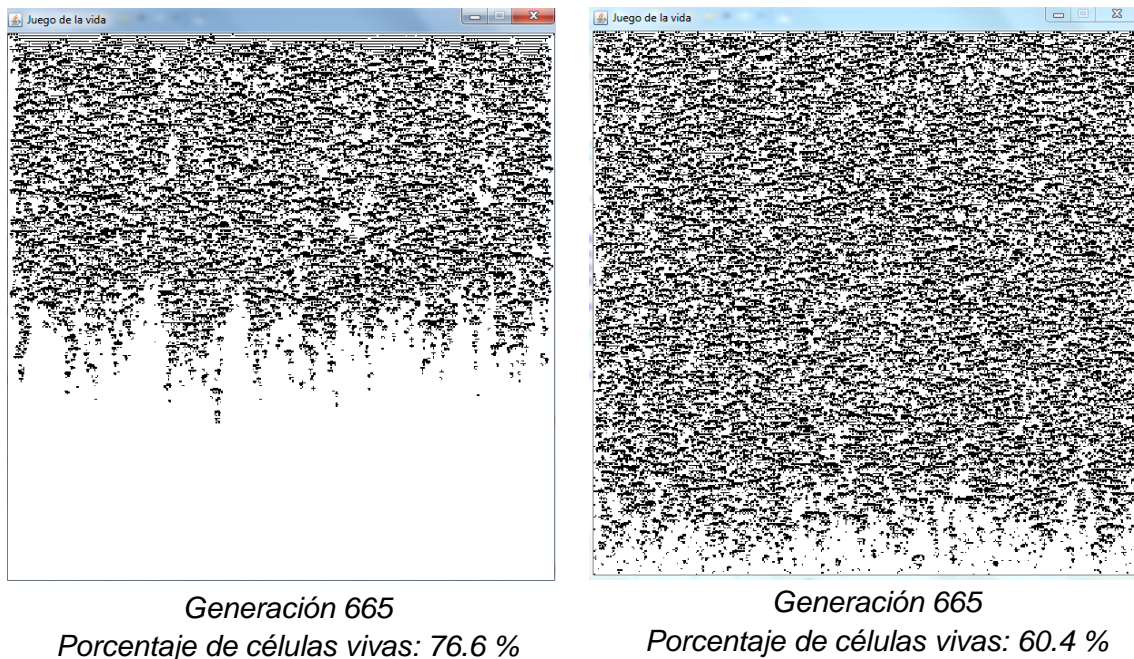


**Figura 4.14.** Representación de la matriz esperada según el JLV y la matriz obtenida al modificar el pvv y pmv.



#### 4.4.3. Variación de probabilidades pvv y pmv para la simulación del apartado 5.4.1

Con el mismo tejido empleado en el apartado 4.4.1, se van a modificar los parámetros pvv y pvm, asignándoles los valores 0,98 y 0,001 respectivamente. Se aprecia un comportamiento diferente al anterior, en el que prevalece la progresión celular, sin embargo, se reduce la velocidad de proliferación del neotejido. Pequeños cambios en estos parámetros pueden dar lugar a grandes variaciones en los resultados.



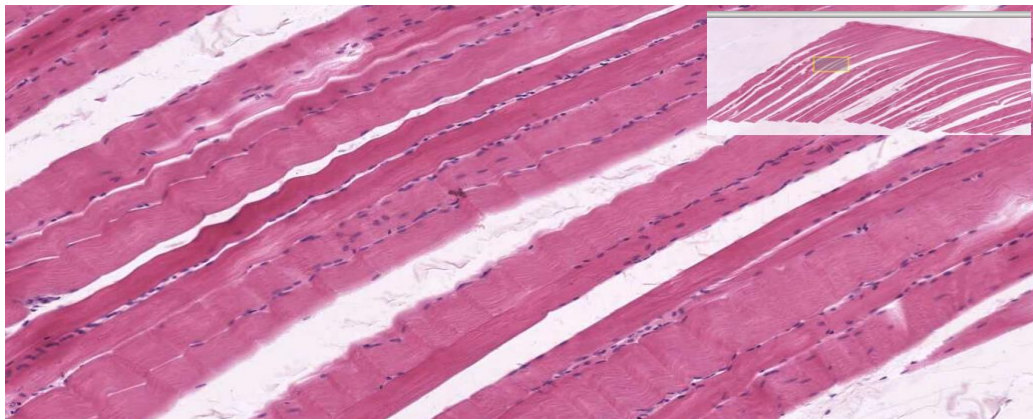
**Figura 4.15.** Representación de un mismo tejido con diferentes parámetros de entrada.

Determinados tratamientos antitumorales, como la quimioterapia o radioterapia, en ocasiones no permiten extinguir el cáncer y son aplicadas con carácter paliativo. Es decir, su misión es intentar ralentizar la progresión tumoral para evitar sintomatología o reducir la invasión que genera el tumor. Probablemente, esta sea otra de las posibles líneas futuras de este TFG, dado que la simulación del efecto de determinados agentes externos con fin terapéutico podría modificar los parámetros del modelo. Por tanto, la capacidad pronóstica simulada del autómatas, podría predecir la efectividad o no de determinados tratamientos.

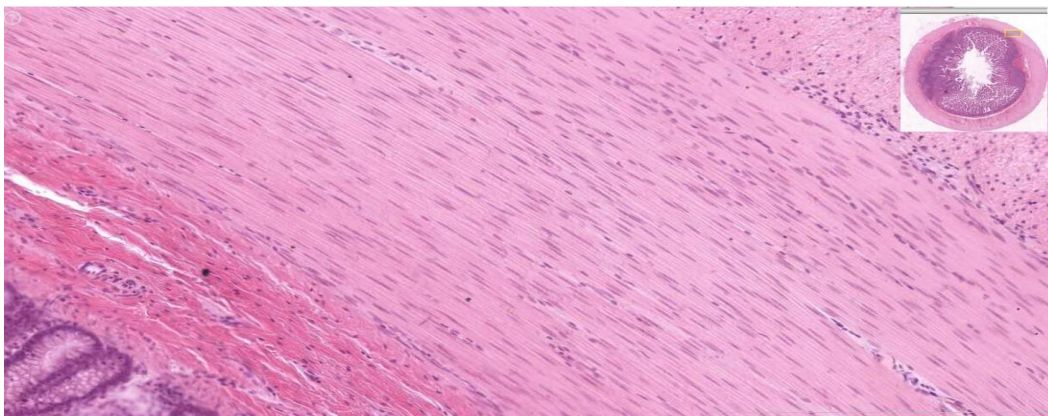
#### 4.5. Simulación 5: Simulación ideal de tejidos con disposición fibrilar.

En esta sección se va a tratar de simular de forma ideal tejidos, cuya disposición es fibrilar. Este puede ser el caso del tejido muscular o del tejido conectivo fibroso. La disposición en estos casos suele estar muy relacionada con la propia función contráctil o de sostén en el organismo. Aunque de manera ideal las simulaciones se realizarán con carácter lineal, se debe subrayar que se continúa representando modelos teóricos, dado que determinadas estructuras fibrilares pueden presentar diferentes geometrías concéntricas o curvilíneas, entre otras.

Si observamos estos tejidos al microscopio óptico, se puede observar en la mayoría de los casos la siguiente distribución por parte de las células:



**Figura 4.16** [19]: Visión de tendón a través de microscopía óptica.



**Figura 4.17** [20]: Visión de tejido muscular liso de una sección del intestino delgado a través de microscopía óptica.

En las dos figuras anteriores, se pueden visualizar las células del tejido, teñidas mediante hematoxilina-eosina, permitiendo diferenciar los componentes celulares. Tal y como se muestra, estas se presentan en una disposición que podríamos considerar lineal.

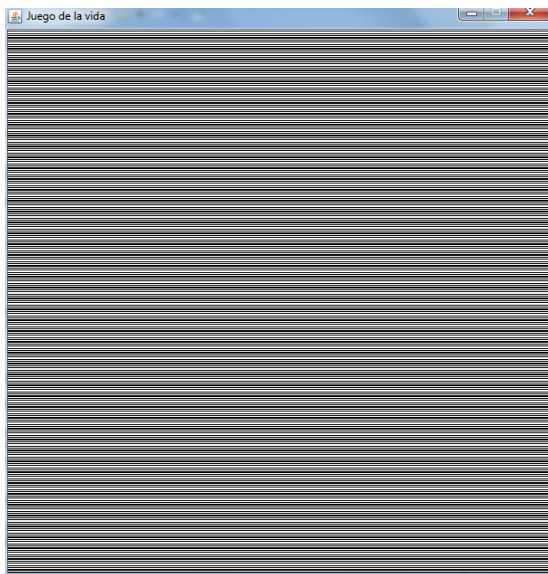
Se ha tratado de simular de manera teórica un tejido que simule esta disposición celular a lo largo del tejido.

A la hora de programar la simulación en este tejido, se han aumentado las probabilidades que determinan que el crecimiento celular sea mayor hacia los laterales, para que simule un crecimiento lineal en el mismo sentido de la fibra. A su vez, se han modificado las reglas, haciendo que estas sean menos restrictivas, para lograr que la simulación fuera coherente. En la figura 4.18 se representa dicha simulación, en la que se puede apreciar un comportamiento anormal. Los parámetros empleados en la simulación se recogen en la tabla 4.4.

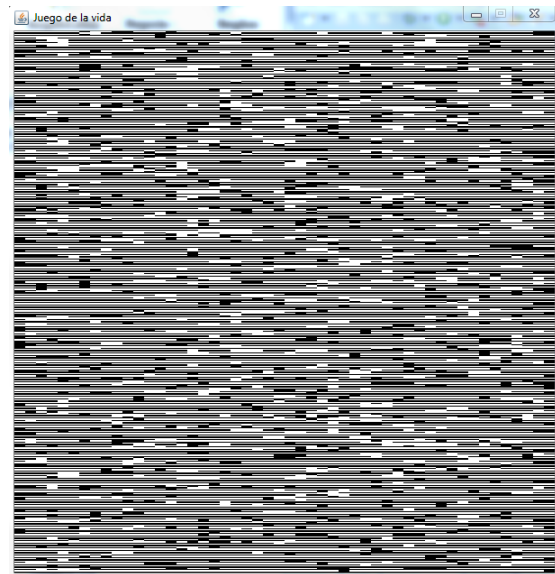
Parámetro	Valor
mMenor	1
mMayor	7
cInferior	6
cSuperior	7
pvv	0.986
pmv	0.0
pv	{0.89, 0.0001, 0.89, 0.98, 0.98, 0.89, 0.0001, 0.89}
pm	{0.0, 0.1, 0.0, 0.0, 0.0, 0.0, 0.1, 0.0}

**Tabla 4.4.** Parámetros empleados para la simulación de la figura 4.18.

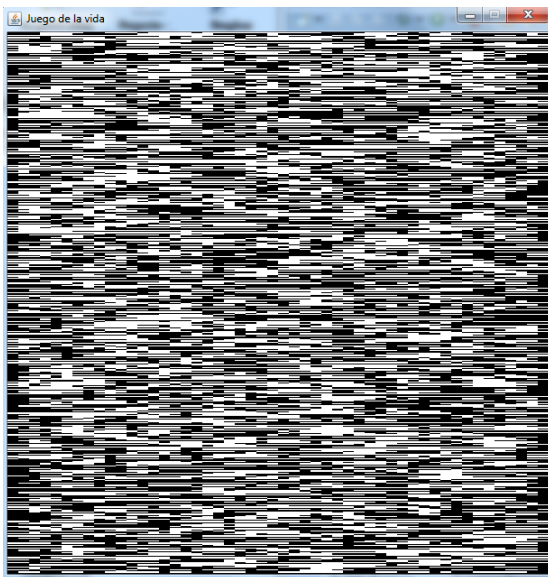
Alterando diferentes parámetros, podemos estudiar comportamientos diferentes del autómatas celular. Para este ejemplo, se puede ver que en la primera generación hay un gran predominio de células vivas. Conforme van avanzando las generaciones, las células vivas se van agrupando, dando lugar finalmente a una sobrepoblación que hace que las células mueran y en consecuencia, el resto de células vayan muriendo por soledad.



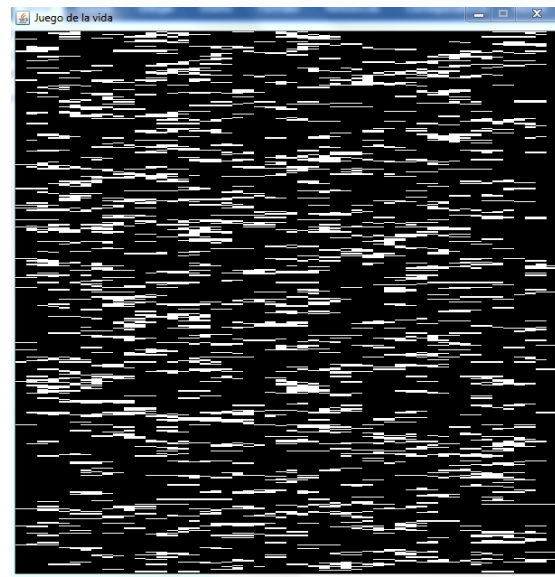
*Generación 1*  
*Porcentaje de células vivas: 49.9%*



*Generación 13*  
*Porcentaje de células vivas: 49.2%*



*Generación 50*  
*Porcentaje de células vivas: 32.85%*



*Generación 105*  
*Porcentaje de células vivas: 12.7%*

**Figura 4.18.** Simulación ideal del comportamiento de un tejido fibroso.

#### 4.6. Simulación 6: Tejido arbitrario con tendencia a disposición diagonal.

En la siguiente simulación se ha empleado un tejido arbitrario cuya dimensión es de 500 x 500. Los parámetros usados están recogidos en la tabla 4.5.

Parámetro	Valor
mMenor	2
mMayor	4
clInferior	3
cSuperior	3
pv	{0.002, 0.8, 0.96, 0.78, 0.78, 0.96, 0.8, 0.002}
pm	{0.0, 0.0015, 0.03, 0.001, 0.001, 0.03, 0.0015, 0.0}

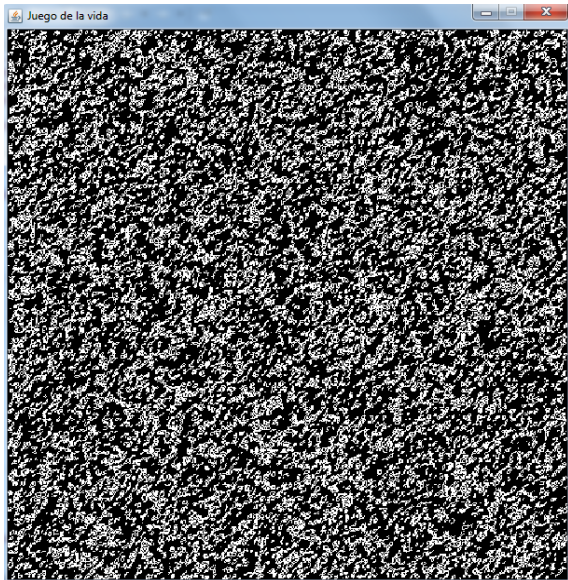
**Tabla 4.5.** Parámetros empleados para la simulación de la figura 4.19. Los datos que no se especifican en la tabla son los mismos al JLV.

En la figura 4.19 se puede observar el resultado de la simulación. Están representadas cuatro de sus generaciones. Como se puede apreciar, conforme se desarrolla el tejido las células tienen a posicionarse de forma diagonal.

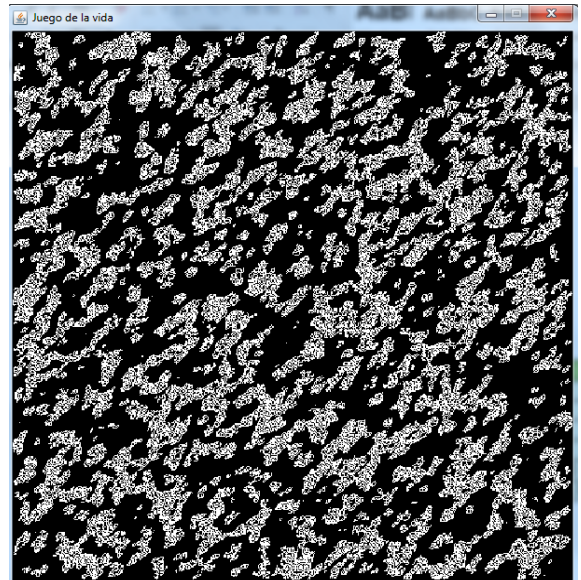
Probando diferentes parámetros se ha llegado a la conclusión de que la configuración más estable para conseguir esta forma de agrupación es estableciendo las reglas y probabilidades determinadas en la tabla 4.5.

La alteración de algunas de los parámetros puede dar lugar a resultados muy diferentes en la evolución del tejido.

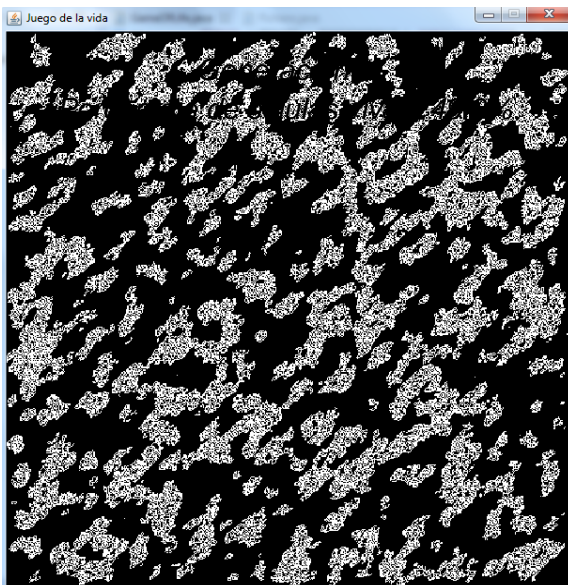




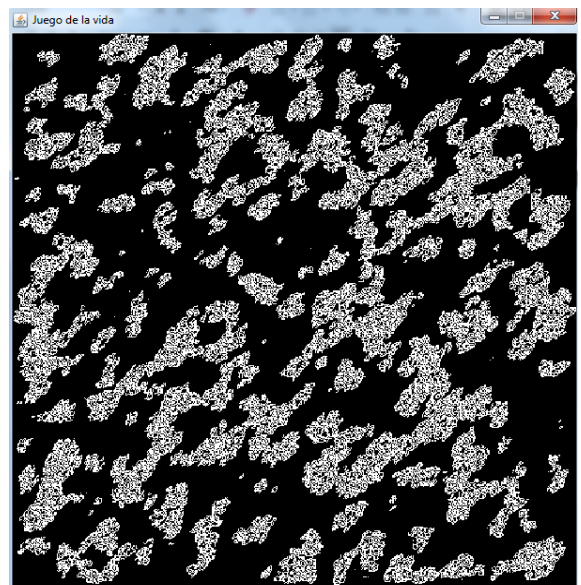
*Generación 3*  
*Porcentaje de células vivas: 32.3%*



*Generación 50*  
*Porcentaje de células vivas: 28.16%*



*Generación 90*  
*Porcentaje de células vivas: 26.47%*



*Generación 160*  
*Porcentaje de células vivas: 24.4%*

**Figura 4.19.** Simulación de tejido arbitrario con tendencia diagonal.

# Capítulo 5: Conclusiones y trabajos futuros.

## 5.1. Conclusiones

A lo largo de la memoria se han ido desarrollando los objetivos de este TFG hasta completarlos.

En primer lugar, se ha realizado una introducción bibliográfica sobre aspectos fundamentales del cáncer, explicando su origen y posterior evolución, grado y estadios que presenta un tumor y sus posibles terapias. También se han desarrollado conceptos básicos relacionados con el JLV, como la descripción del autómatas celular, ejemplos de funcionamiento, patrones celulares, entre otros.

En segundo lugar, se ha realizado una aplicación basada en el JLV de John Horton Conway. Este es un programa más versátil ya que emplea autómatas celulares probabilísticos. Este dispone de más parámetros que rigen la evolución de dicho autómatas, incrementándose las posibilidades de respuesta, y por ello se incrementa la complejidad de los modelos, de manera que pueden llegar a emular situaciones reales.

En tercer lugar, se ha realizado una fase de experimentación simulando tejidos de diferentes dimensiones y probando distintos parámetros para examinar que clase de comportamientos adoptaba el autómatas en cada situación. Con los resultados obtenidos, se ha tratado de dar una correspondencia con procesos fisiopatológicos que acontecen en el cuerpo humano. En los mismos, se usan a modo de ejemplo el comportamiento de diferentes tumores, así como el crecimiento de otros tejidos el organismo, de manera que se facilita la comprensión y la aplicabilidad de las diferentes simulaciones generadas.

Con este TFG se puede apreciar la capacidad que puede llegar a tener un autómatas celular probabilístico, dado que, a partir de la imposición de unas

reglas sencillas, se pueden llegar a modelar comportamientos complejos. A día de hoy, es complicado predecir la evolución que puede tener un tumor. Aunque la implementación actual de este trabajo se ha realizado para el estudio de tejidos arbitrarios y de forma genérica, podría ser el nexo de unión para que en futuros trabajos pudieran llegarse a modelar un determinado comportamiento esperable de un conjunto de células cancerígenas y poder caracterizar de forma individualizada cada tejido. Esto podría suponer un avance a nivel clínico para diagnosticar y predecir el comportamiento de neoplasias.

## **5.2. Futuras líneas de desarrollo**

Dado que este TFG tiene un carácter preliminar y básico, como futuros trabajos cabría destacar las siguientes mejoras:

- Implementar en tres dimensiones el programa. Sería mucho más realista que la implementada actualmente en dos dimensiones, dado que esta es la configuración real del cuerpo humano.
- Caracterización de manera específica determinados tejidos de interés que se deseen examinar, así como las reglas que rigen la evolución del autómata celular probabilístico para cada componente tisular. De esta forma, se podría llegar a saber el comportamiento de estos tejidos ante determinadas situaciones.
- Caracterizar el comportamiento que sigue un tejido cuando se le aplican terapias para extinguir o paliar el tumor. De esta forma, se podrían generar diferentes modelos de respuesta de los tejidos ante determinados agentes externos, como pueden ser los tratamientos dirigidos para modular o tratar procesos neoplásicos.
- Distinción entre células normales y células cancerígenas.
- Hacer que las probabilidades que rigen al autómata celular probabilístico no sean fijas, sino que sigan una distribución de probabilidad.
- Asignación de probabilidades distintas sobre las células que mueren, en



función de las causas de la muerte, ya sea por aislamiento o sobrepoblación.

Partiendo de este TFG, se podrían plantear otra serie de trabajos futuros fuera el campo de la oncología.

Existen determinados fenómenos que alteran la homeostasis del tejido. Este hecho acontece cuando falta riego sanguíneo durante un infarto de corazón o ictus cerebral. Este cambio en el patrón tisular genera una modificación en las variables del entorno celular, que pudieran ser parametrizadas mediante este modelo probabilístico.

Por otro lado, la senescencia celular programada es algo inevitable, sobre el que se diseñan determinadas terapias que intentan reducir el estrés oxidativo celular, entre otras. Aunar estas líneas de investigación con modelos como el de este TFG podría ayudar a optimizar el conocimiento en dicho campo.

Por último, subrayar que la traslación del campo de la ingeniería, matemática y estadística, hasta el ámbito de la salud, permitirá un abordaje multidisciplinar en las enfermedades. Hecho que será clave en el futuro en el conocimiento médico.



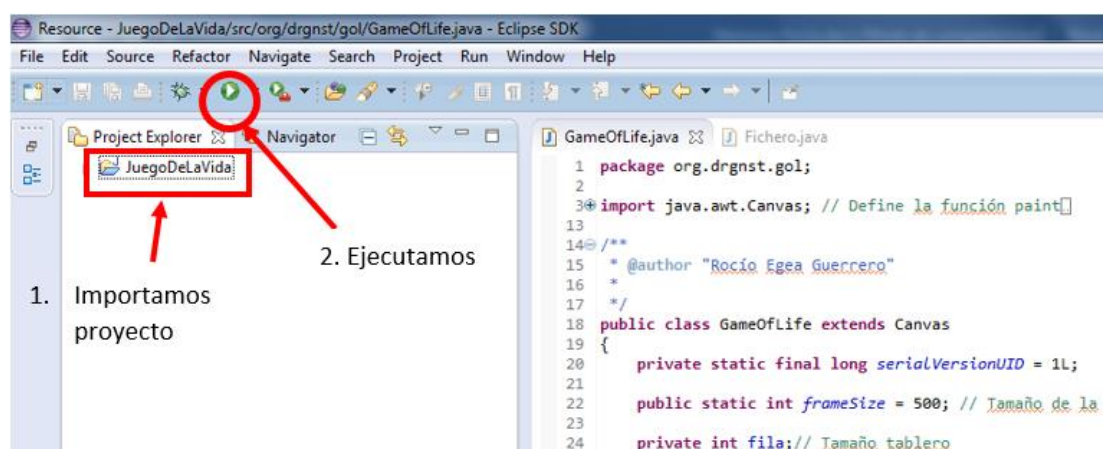
# Anexo A: Manual de usuario

## A.1. Instalación y puesta en marcha

En el CD que incorpora la memoria, hay dos archivos dentro de la carpeta *Simulación*. En ella podemos encontrar el proyecto “JuegoDeLaVida” y “AppJuegoVida.jar”. Por lo tanto, existen dos maneras de ejecutar este programa.

1. Si se decide ejecutar el proyecto “JuegoDeLaVida” es necesario tener instalado el entorno de desarrollo que se ha empleado para la realización del programa, en este caso Eclipse. En el caso de no disponer de este programa, se puede obtener gratuitamente desde la página oficial de eclipse. [14].

Una vez abierto Eclipse, se importa el proyecto y se pincha sobre la tecla Run.



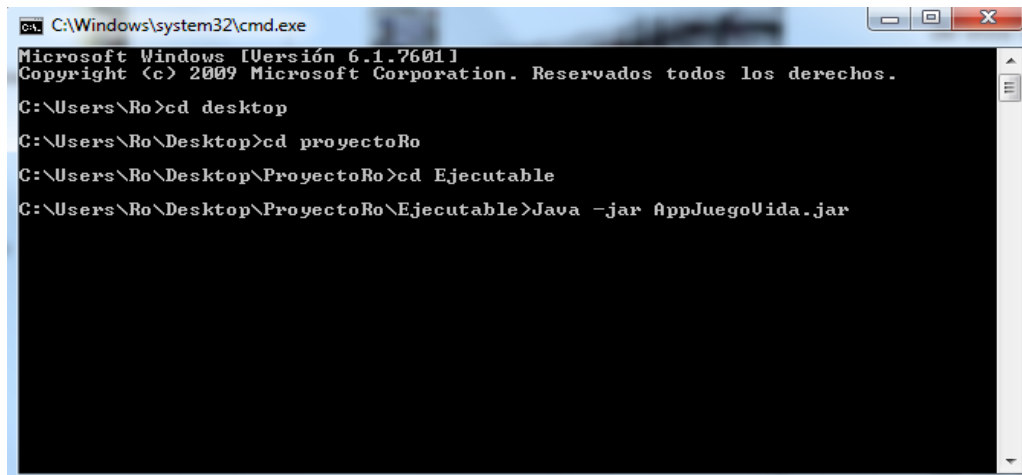
**Figura A.1.** Ejecución del programa mediante Eclipse.

2. La segunda alternativa es ejecutar “AppJuegoVida.jar”. Para ello es necesario usar la consola de comandos del ordenador (cmd).

Desde el cmd se debe de ir al directorio donde se ha ubicado el archivo.jar. Para ello se usa el comando Cd <<nombre directorio>>. Si el archivo se encontrara en el escritorio habría que poner simplemente “Cd Desktop”. En el caso de que el archivo estuviera guardado dentro de una carpeta del escritorio

o subcarpetas, simplemente habría que llamar de nuevo a Cd con el nombre de la carpeta que lo contiene de forma sucesiva.

Una vez en el directorio correcto, se ejecuta la aplicación con el siguiente comando: Java -jar AppJuegoVida.jar



```
C:\Windows\system32\cmd.exe
Microsoft Windows [Versión 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. Reservados todos los derechos.

C:\Users\Ro>cd desktop
C:\Users\Ro\Desktop>cd proyectoRo
C:\Users\Ro\Desktop\ProyectoRo>cd Ejecutable
C:\Users\Ro\Desktop\ProyectoRo\Ejecutable>Java -jar AppJuegoVida.jar
```

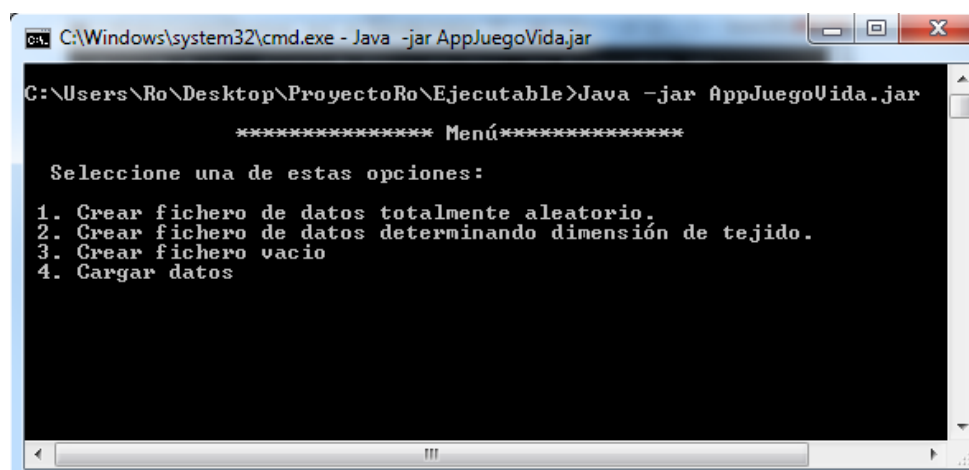
*Figura A.2. Ejecución del programa mediante cmd.*

Una vez escogida una de las dos alternativas y realizado los puntos

explicados anteriormente, los siguientes pasos son iguales para ambos casos. Por lo tanto, se va a continuar explicando el proceso mediante la consola de comandos.

## A.2. Entrada de datos

En primer lugar, el programa nos va a mostrar el siguiente menú:



```
C:\Windows\system32\cmd.exe - Java -jar AppJuegoVida.jar
C:\Users\Ro\Desktop\ProyectoRo\Ejecutable>Java -jar AppJuegoVida.jar

***** Menú*****

Seleccione una de estas opciones:

1. Crear fichero de datos totalmente aleatorio.
2. Crear fichero de datos determinando dimensión de tejido.
3. Crear fichero vacío
4. Cargar datos
```

*Figura A.3. Menú inicial del programa.*

### A.2.1. Creación de datos totalmente aleatoria

Si se pulsa la tecla 1, el programa en primer lugar preguntará que nombre se le desea poner al nuevo fichero. Una vez introducido, también pregunta cómo quiere que genere los datos de probabilidad, si por defecto, los cuales corresponden a los del JLV original o bien aleatorios. Por último, a partir de la información aportada al programa, se dispondrá a crear el fichero de datos.

```
C:\Users\Ro\Desktop\ProyectoRo\Ejecutable>java -jar appjuegoUida.jar

***** Menú*****

Seleccione una de estas opciones:

1. Crear fichero de datos totalmente aleatorio.
2. Crear fichero de datos determinando dimensión de tejido.
3. Crear fichero vacio
4. Cargar datos

1

Por favor introduzca el nombre del fichero que quiera analizar :
prueba1

¿Quiere usar probabilidades por defecto <0> o aleatorias <1>?
0
```

*Figura A.4. Apariencia de la primera opción de entrada de datos*

### A.2.2. Creación de fichero de datos determinando la dimensión del tejido

Si se pulsa la tecla 2, el programa, al igual que se ha descrito en el apartado anterior, preguntará que nombre se le desea poner al nuevo fichero. Una vez introducido, pedirá que se introduzca las dimensiones deseadas del tejido y si se quiere las probabilidades por defecto o aleatorias. Para finalizar, a partir de la información aportada el programa, se creará el fichero de datos.

```
***** Menú*****

Seleccione una de estas opciones:

1. Crear fichero de datos totalmente aleatorio.
2. Crear fichero de datos determinando dimensión de tejido.
3. Crear fichero vacío
4. Cargar datos

2

Por favor introduzca el nombre del fichero que quiera analizar :
prueba2

Introduzca el tamaño del tejido que quiera analizar.

Fila:
10
Columna:
10

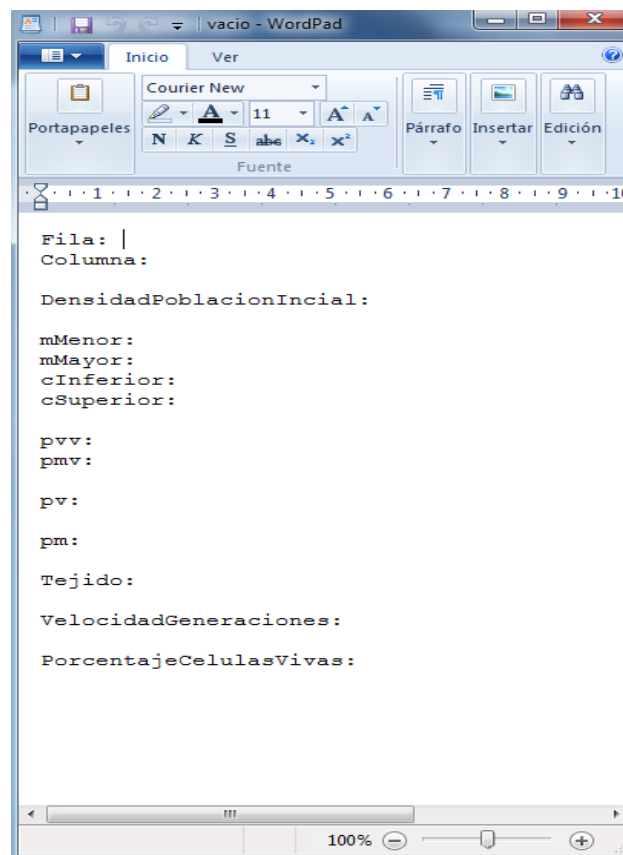
¿Quiere usar probabilidades por defecto <0> o aleatorias <1>?

1
```

*Figura A.5. Ejemplo demostrativo de opción dos de entrada de datos.*

### A.2.3. Creación de fichero de datos vacío

Al pulsar la tecla 3, el programa nos genera un fichero de texto vacío como muestra la figura A.6 a partir de un nombre que el usuario introduce por consola.



*Figura A.6. Fichero vacío generado*

El objetivo de esta alternativa es que el usuario pueda introducir los datos de forma manual a partir de un fichero de texto, el cual, se guardará de manera automática en la misma carpeta donde se encuentre “AppJuegoVida.jar”. Si el usuario elige esta opción, deberá rellenar el fichero con los datos de interés para su estudio, como en el ejemplo de la figura A.7 (El tipo de dato que debe ser escrito en cada sección está indicado en el apartado 3.2.1). Posteriormente, se volverá a ejecutar el programa y se seleccionará la opción 4, debido a que, como se explicará en el apartado 4.2.4., es la encargada de leer ficheros ya existentes.

```
Fila: 10
Columna: 10

DensidadPoblacionInicial: 0.35

mMenor: 2
mMayor: 3
cInferior: 3
cSuperior: 3

pvv: 0.54
pmv: 0.73

pv:
0.06 0.09 0.39 0.17 0.81 0.8 0.45 0.65

pm:
0.12 0.8 0.89 0.4 0.74 0.9 0.59 0.47

Tejido:
1 0 1 0 1 1 0 1 0 1
1 1 1 1 1 0 0 0 0 0
1 0 1 0 1 0 0 0 0 0
1 1 1 1 0 1 1 0 0 0
1 0 1 1 1 1 0 0 0 0
0 0 0 0 0 0 0 1 1 0
0 0 0 0 1 1 0 0 0 1
0 0 0 0 0 0 1 0 1 0
1 0 1 0 0 1 0 1 0 1
1 0 0 1 0 1 0 1 1 1

VelocidadGeneraciones: 1.0

PorcentajeCelulasVivas: 0.43
```

**Figura A.7.** Fichero de ejemplo

El aspecto de la consola en este caso sería el siguiente:

```
***** Menú*****  
  
Seleccione una de estas opciones:  
1. Crear fichero de datos totalmente aleatorio.  
2. Crear fichero de datos determinando dimensión de tejido.  
3. Crear fichero vacío  
4. Cargar datos  
  
3  
  
Por favor introduzca el nombre del fichero que quiera analizar :  
vacío  
  
Fichero creado con éxito.
```

*Figura A.8.: Ejemplo de tercera opción del menú de inicio*

#### A.2.4. Cargar datos

Si se elige la cuarta opción es debido a que ya se tiene un fichero con los datos de interés para el estudio, ya sean los ficheros que han sido creados generación tras generación por el programa en estudios anteriores, ficheros generados por el programa anteriormente que se quieren volver a examinar o aquellos que desde un fichero vacío (apartado 4.2.3) se hayan completado.

Es importante destacar que los ficheros deben de estar guardado en la misma carpeta donde se encuentra “AppJuegoVida.jar” para que el programa pueda encontrarlos y ejecutarlos.

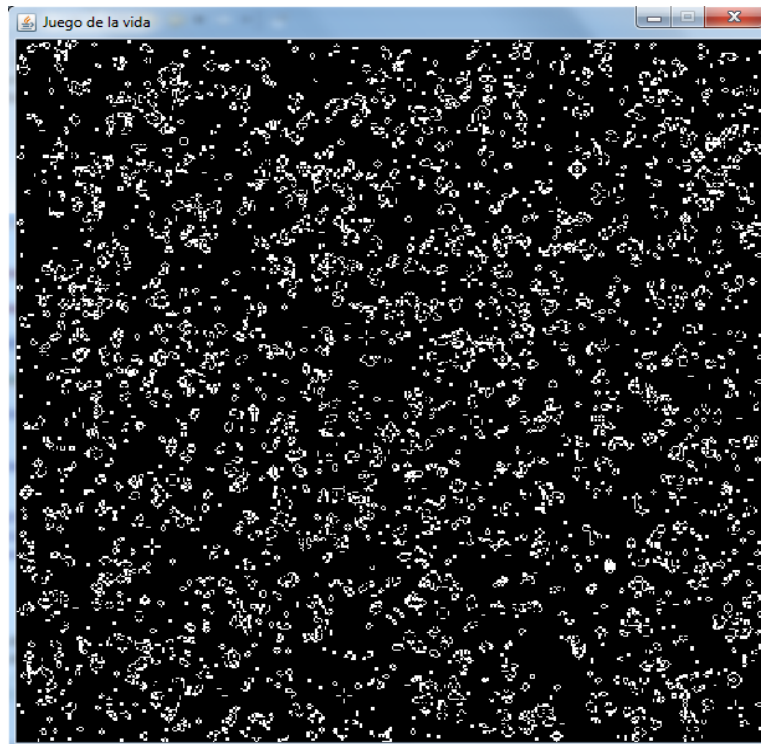
```
***** Menú*****  
  
Seleccione una de estas opciones:  
1. Crear fichero de datos totalmente aleatorio.  
2. Crear fichero de datos determinando dimensión de tejido.  
3. Crear fichero vacío  
4. Cargar datos  
  
4  
  
Por favor introduzca el nombre del fichero que quiera analizar :  
prueba2
```

*Figura A.9. Ejemplo de cuarta opción del menú de inicio*



### A.3. Interfaz gráfica de resultados

Como resultado, se obtiene una ventana que muestra los resultados del comportamiento del autómata celular en función de los parámetros introducidos.



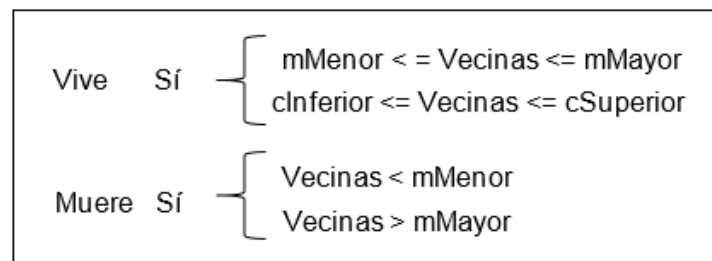
**Figura A.10.** Ejemplo de interfaz gráfica



## Anexo B. Interpretación gráfica de la variación de las reglas del autómata celular.

En la figura B.1, se representan las condiciones, de forma genérica, que marcan la supervivencia, nacimiento o muerte de una célula para la siguiente generación, en función de los parámetros que lo determinan.

Condiciones de vida y muerte:

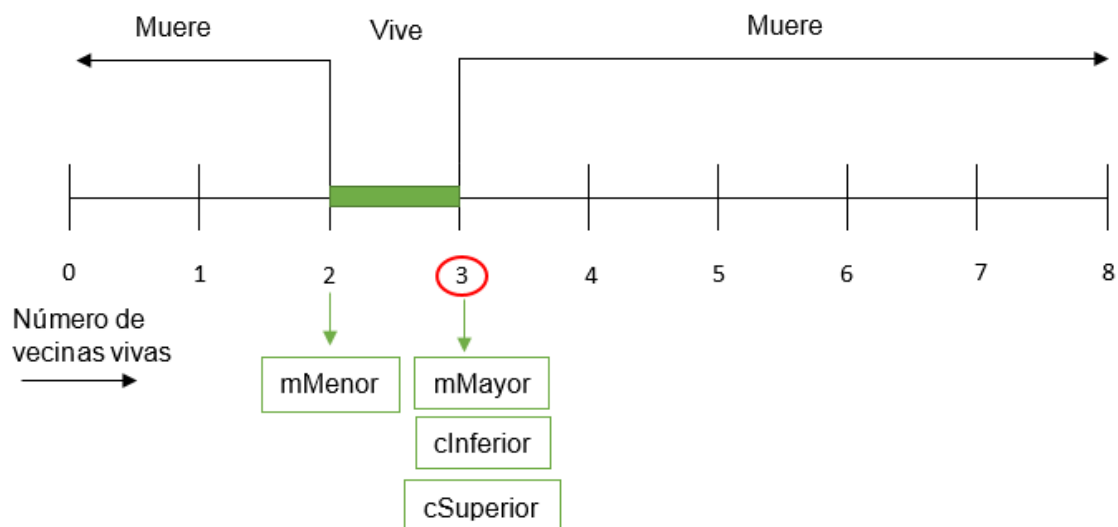


**Figura B.1:** Condiciones genéricas que determinan el estado de la célula en estudio para la siguiente generación.

En la figura B.2, a partir de estas condiciones, se ha realizado una representación gráfica con los parámetros que caracterizan el caso concreto del JLV original. El eje horizontal representa el número de células vivas que puede tener la célula en estudio. La zona representada en color verde, simboliza el intervalo donde se encontrarán vivas las células en la siguiente generación, mientras que, la zona sin colorear representa el rango en el que las células estarán muertas. El círculo rojo simboliza el número de células vecinas que deben de estar vivas alrededor de la célula de estudio en el caso de que esté muerta, para que nazca en la siguiente generación.

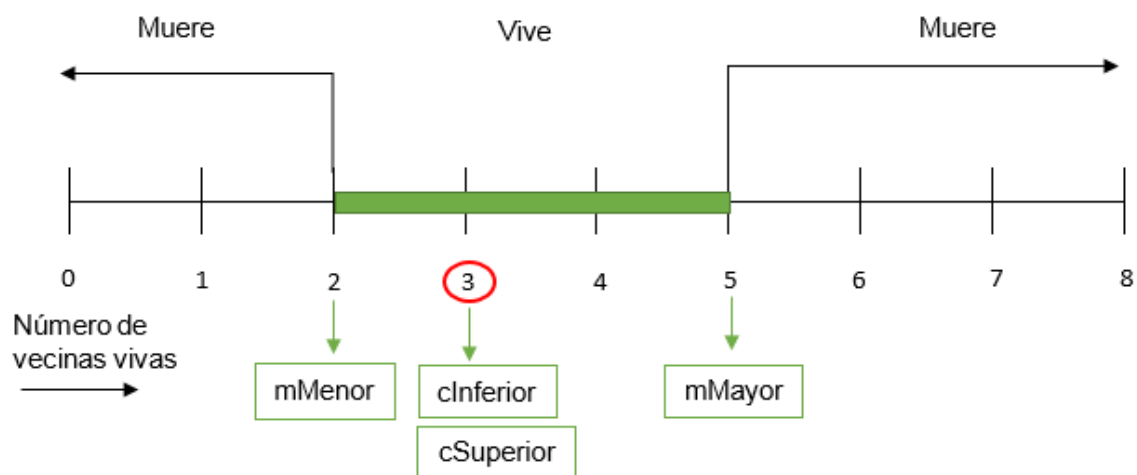
Si se cambian los parámetros que determinan las reglas, también cambiará el comportamiento del autómata celular. En la figura B.3 se puede comprobar que si se aumenta el valor de  $mMayor$  habrá más espectro de células vivas que en la figura B.2, como se puede observar en el área verde de las dos figuras.

Representación de condiciones:



**Figura B.2.** Representación gráfica de las condiciones que determinan el desarrollo del autómat celular para el JLV original (Simulación A, figura 4.9).

Representación de condiciones:



**Figura B.3.** Representación gráfica de las condiciones impuestas en la simulación C de la figura 4.9.

En conclusión, variando los parámetros que determinan las reglas se puede conseguir que las células del tejido se comporten de una determinada manera. Cuanto menos restrictivas sean las reglas, más predominio de células vivas habrá en el tejido, las cuales serán más resistentes y permanecerán más tiempo en este. Si en caso contrario, se imponen unas reglas muy restrictivas, será más difícil para las células sobrevivir o expandirse de manera descontrolada.



# Glosario de términos

- **Apoptosis celular:** Es la muerte celular programada ante estímulos extra o intracelulares.

- **Autómata celular:** Sistema dinámico formado por un conjunto de elementos sencillos idénticos entre sí, pero que en conjunto son capaces de demostrar comportamientos complejos. El estado de cada elemento depende del estado previo de los elementos vecinos, según un conjunto de reglas de transición.

- **Cáncer:** Enfermedad que se caracteriza por la transformación de las células, que proliferan de manera anormal e incontrolada.

- **Célula:** Unidad mínima fundamental de los organismos vivos con capacidad de reproducción independiente.

- **Hematoxilina-eosina:** Técnica de tención utilizada en histología, la cual se emplea para observar al microscopio óptico ciertos componentes de un tejido.

- **Necrosis:** Es la muerte de las células de un tejido que puede ser originado por diferentes factores, como falta de irrigación, radiación entre otras.

- **Neoplasia:** Crecimiento descontrolado de células en uno o varios tejidos del organismo. Sinónimo de la palabra tumor.

- **Patrón:** Sucesión de elementos que se construye siguiendo una regla y dichos elementos se representan de forma periódica.

- **Senescencia celular:** Cambio irreversible de las células producida por el envejecimiento como respuesta al estrés o daño de estas.

- **Tumor:** Masa de células transformadas, con crecimiento y multiplicación anormales.





# Biografía y referencias

- [1] Romero Dopico M. Universidad Carlos III de Madrid (No date). El JLV. Disponible en:  
<http://www.it.uc3m.es/jvillena/irc/practicas/09-10/04mem.pdf>
- [2] Acera M. (2013). El JLV para estudiantes de programación. Disponible en:  
<http://miquelangelacera.com/2013/01/21/el-juego-de-la-vida-para-estudiantes-de-programacion/>
- [3] Organización mundial de la salud (2015). Cáncer. Disponible en:  
<http://www.who.int/mediacentre/factsheets/fs297/es/>
- [4] José Luis Galán García, Gabriel Aguilera Venegas, Pedro Rodríguez Cielos: An accelerated-time simulation for traffic flow in a smart city. J. Computational Applied Mathematics 270: 557-563 (2014).
- [5] José Luis Galán García, Gabriel Aguilera Venegas, María Ángeles Galán García, Pedro Rodríguez Cielos: A new Probabilistic Extension of Dijkstra's Algorithm to simulate more realistic traffic flow in a smart city. Applied Mathematics and Computation 267: 780-789 (2015).
- [6] Gabriel Aguilera Venegas, José Luis Galán García, Enrique Mérida Casermeiro, Pedro Rodríguez Cielos: An accelerated-time simulation of

baggage traffic in an airport terminal. Mathematics and Computers in Simulation 104: 58-66 (2014).

- [7] Gabriel Aguilera Venegas, José Luis Galán García, José Manuel García, Enrique Mérida Casermeiro, Pedro Rodríguez Cielos: An accelerated-time simulation of car traffic on a motorway using a CAS. Mathematics and Computers in Simulation 104: 21-30 (2014).
- [8] Valladares Sánchez, Ysidro (2007). Historia natural del cáncer. Causas, efectos y manifestaciones.
- [9] Instituto nacional del cáncer. Disponible en:  
<https://www.cancer.gov/espanol/cancer/diagnostico-estadificacion/pronostico/hoja-informativa-grado-tumor>
- [10] Asociación española contra el cáncer. Disponible en:  
<https://www.aecc.es/SobreElCancer/Tratamientos/Paginas/Tratamientos.aspx>
- [11] Economía digital. Smoothlife: El JLV en versión continua con valores de coma flotante. Disponible en:  
<http://www.microsiervos.com/archivo/juegos-y-diversion/smoothlife-juego-de-la-vida.html>
- [12] Reyes Gómez D, Descripción y aplicaciones de los Autómatas celulares. (2011). Disponible en:

[http://delta.cs.cinvestav.mx/~mcintosh/cellularautomata/Summer\\_Research\\_files/Arti\\_Ver\\_Inv\\_2011\\_DARG.pdf](http://delta.cs.cinvestav.mx/~mcintosh/cellularautomata/Summer_Research_files/Arti_Ver_Inv_2011_DARG.pdf)

- [13] Andrew Adamatzky. Springer-Verlag London (2010). Game of Life Cellular Automata.
- [14] Entorno de desarrollo Eclipse.\_Disponible en:  
<https://eclipse.org/downloads/packages/release/Juno/SR2>
- [15] Microscopio virtual. Universidad de Málaga. Tejido hepático. Disponible en:  
<https://dj.uma.es/microscopio/gallery/308/>
- [16] Microscopio virtual. Universidad de Málaga. Tejido Cartilaginoso. Disponible en:  
<https://dj.uma.es/microscopio/gallery/834/>
- [17] Imagen de tumor pulmonar con necrosis. Disponible en:  
<http://www2.udec.cl/~fmucien/sitio2/02.%20Patologia%20Pulmonar/Patologia%20Pulmonar.html>
- [18] Imagen de tumor con carácter infiltrante. Disponible en:  
<http://www.monografias.com/trabajos105/cancer-colon-el-estigma-oculto/cancer-colon-el-estigma-oculto.shtml>
- [19] Microscopio virtual. Universidad de Málaga. Tejido conectivo fibroso:

Tendón. Disponible en:

<https://dj.uma.es/microscopio/gallery/740/>

- [20] Microscopio virtual. Universidad de Málaga. Tejido muscular liso.

Disponible en:

<https://dj.uma.es/microscopio/gallery/984/>